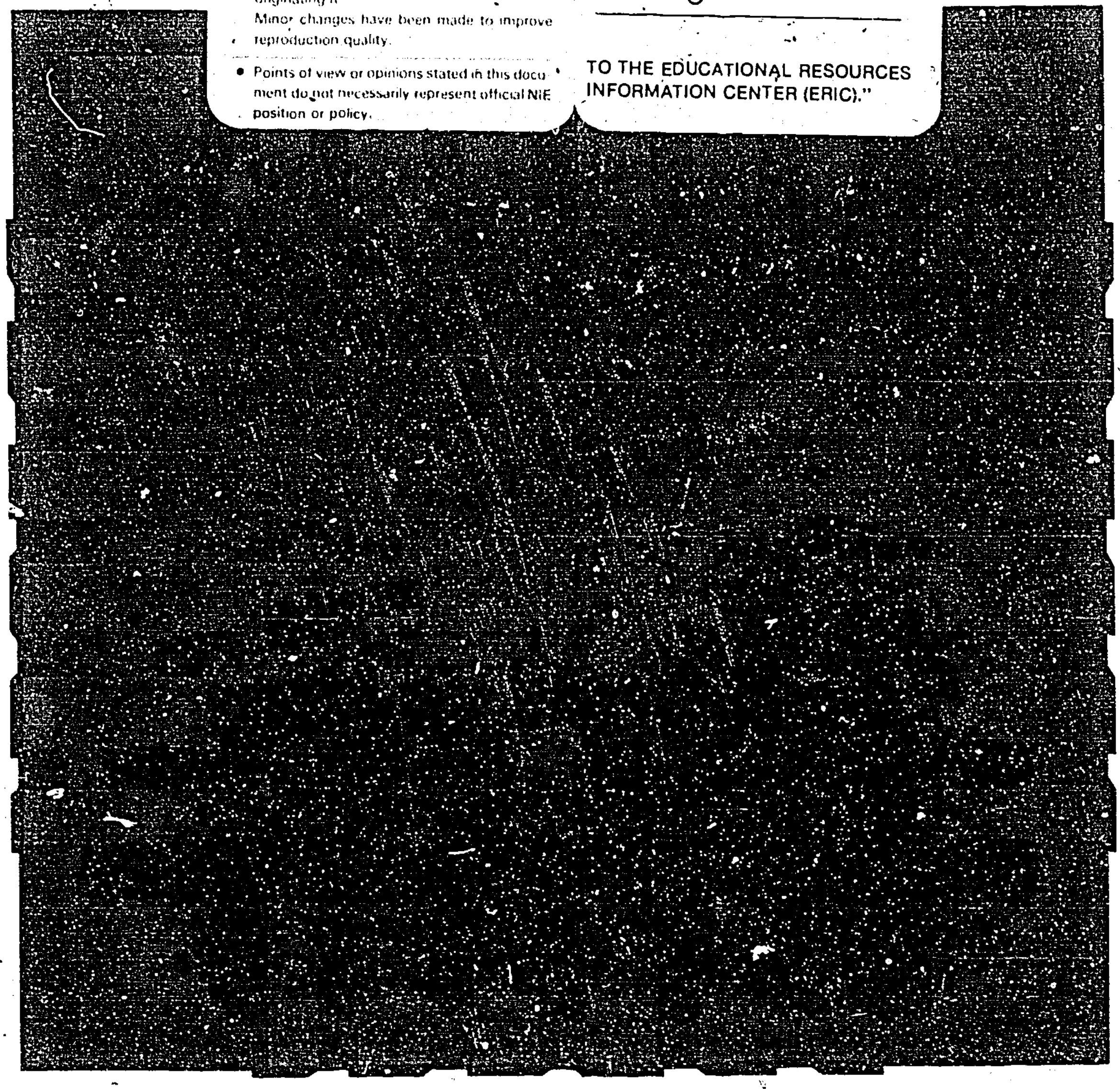
DOCUMENT RESUME

ED 239 397 EA 016 358

TITLE	The Computer: Extension of the Human Mind II. Proceedings of the Annual Summer Conference (Eugene, Oregon, July 20-22, 1983).
INSTITUTION	Oregon Univ., Eugene. Coll. of Education.
PUB DATE	83
NOTE	239p.
AVAILABLE FROM	Publications, Summer Conferences, Office of the Dean, College of Education, University of Oregon, Eugene, OR 97403 ($10.00 prepaid or purchase order; make checks payable to Summer Conferences).
PUB TYPE	Collected Works - Conference Proceedings (021)
EDRS PRICE	MF01/PC10 Plus Postage.
DESCRIPTORS	Business Education; Computational Linguistics; *Computer Assisted Instruction; Computer Graphics; Computer Literacy; *Computer Oriented Programs; *Computers; Computer Science Education; Curriculum Development; Educational Administration; *Educational Technology; Elementary Secondary Education; *Microcomputers; Special Education

ABSTRACT

This collection consists of 21 papers presented at the July 1983 Annual Summer Conference on "The Computer: Extension of the Human Mind II," in Eugene, Oregon. Six papers were presented at general interest sessions; 15 were from special interest group sessions. The general interest papers include David Ahl's "Keeping Up with Computers in Education or Computer Periodicals: Past, Present, and Future"; Kenneth Komoski's "The Computer: Extension of the Human Mind and Challenge to Humanness"; J. D. Fletcher's "New Directions for Computer Courseware"; Arthur Luehrmann's "Microcomputers in the Junior High School"; Dorothy Deringer's "Computers in Education: Activities at the Federal Level"; and Kenneth Brumbaugh's "Developing and Distributing Microcomputer Software." The 15 special session papers cover a wide range of topics, including microcomputer applications in such areas as management information systems (Bruer), educational management (Piele), and career information (McKinlay), as well as curriculum applications for microcomputers in secondary school business (Lidtke), writing (Herrman), mathematics (Johnson), fine arts (Jones), elementary level computer science (Arch), physical sciences (Stringer), and special education (Metzger). Other topics include problems in computer graphics (Hill), computerized toys (Moore), videodiscs (Moulton), databases for locating software (Zaporozhetz), and a panel discussion on teacher education in computers (Moursund). (TE)



THE COMPUTER: EXTENSION OF THE HUMAN MIND ANNUAL CONFERENCE

College of Education · University of Oregon

PROCEEDINGS OF THE ANNUAL SUMMER CONFERENCE

THE COMPUTER: EXTENSION OF THE HUMAN MIND II

Sponsored by

COLLEGE OF EDUCATION

University of Oregon

Held at the

EUGENE HILTON AND CONVENTION CENTER

Eugene, Oregon

July 20-22, 1983

PREFACE

The College of Education at the University of Oregon is pleased to offer this printed proceedings of its Second Annual Summer Conference on The Computer: Extension of the Human Mind held in Eugene, Oregon, July 20-22, 1983.

The authors of these papers were invited to make presentations at this conference because of their acknowledged scholarly or professional competence in some aspect of computers and education. The findings, views, or opinions expressed by those authors, however, are entirely their own and do not necessarily represent those of the College of Education or the University of Oregon.

To meet our publication deadline and control printing costs, the papers are presented exactly as they were received; not even the pagination was changed. Only one of the presenters was unable to make the printing deadline.

Additional copies of the proceedings may be obtained by sending $10.00 (prepaid or purchase order) to Summer Conferences, Office of the Dean, College of Education, University of Oregon, Eugene, Oregon 97403. Checks should be made payable to Summer Conferences.

David Moursund
Philip K. Piele
Co--Chairs
Conference Program Committee

CONTENTS

(To meet our publication deadline and control printing costs, the papers are presented exactly as they were received.)

General Interest Sessions

Special Interest Group Sessions

1. Microcomputers and Management Information Systems: An Emerging Partnership
   Leon C. Bruer
   Director of Data Processing
   Linn-Benton Community College
   Albany, Oregon

2. Using Computers in Secondary Business Curriculum
   Doris K. Lidtke
   Professor of Mathematics and Computer Science
   Towson State University
   Baltimore, Maryland

3. Using the Computer as Writing Teacher: The Heart of the Great Debates
   Andrea W. Herrmann
   Doctoral Student in Applied Linguistics and Computer Science
   Teachers College
   Columbia University
   New York, New York

4. Problems in Computer Graphics
   Lee T. Hill
   Professor of Mathematics and Computer Science
   Southern Oregon State College
   Ashland, Oregon

5. Teacher Education for Computers in Secondary School Curriculum: A Panel Discussion
   David Moursund
   Professor of Computer and Information Science
   University of Oregon
   Eugene, Oregon
   With Doris Carey, Regan Carey, and Keith Wetzel
   Graduate Students in Computer and Information Science

6. Thinker Toys
   Craig Moore
   Teacher
   Harriet Tubman Middle School
   Portland, Oregon

7. Videodisc/Microcomputer Systems in Instruction
   Peter Moulton
   Adjunct Assistant Professor of Computer and Information Science
   University of Oregon
   Eugene, Oregon

8. The Use of Computerized Databases to Locate Information on Software
Laurene E. Zaporozhetz
Assistant Professor
Reference Department
University of Oregon Library
Eugene, Oregon

9. Microcomputer Applications of Career Information
Bruce McKinlay
Director
Career Information Systems
College of Education
University of Oregon
Eugene, Oregon

10. Microcomputers in School Science
Gene A. Stringer
Professor of Physics
Southern Oregon State College
Ashland, Oregon

11. Microcomputer Applications in Educational Management: A Review of Integrated Management Software
Philip K. Piele
Professor and Director
ERIC Clearinghouse on Educational Management
College of Education
University of Oregon
Eugene, Oregon

12. A Model for Developing An Elementary School Computer Science Curriculum
John C. Arch
Teaching Fellow
Department of Computer and Information Science
University of Oregon
Eugene, Oregon

13. Mathematics Education and Computers: Cause for Concern or Change?
Jerry Johnson
Professor of Mathematics
University of the Pacific
Stockton, California

14. Computers: A Possible Solution to the Crisis Facing Special Education
Merrianne Metzger
Teacher
Bend/La Pine Public Schools
Bend, Oregon

15. <u>Computer Assisted Instruction and Graphics</u>,

Beverly J. Jones
Assistant Professor of Art Education
University of Oregon
Eugene, Oregon

MICROCOMPUTERS AND MANAGEMENT INFORMATION SYSTEMS: AN EMERGING PARTNERSHIP

Leon C. Bruer

Section 1

Microcomputers - New Performers in the Computer Arena.

When historians have completed their compilation and analysis of the events of the 1980s, there is no doubt that a major share of the changes in human life during the decade will be traceable to the advent of the microcomputer. The growth and unfolding of the potential of this latest computer revolution is happening so quickly that our ability to understand and guide the changes into the most productive directions is severely strained.

Our collective stake in this process is enormous. By various estimates, the microcomputer and information processing industry is growing at a rate of approximately 30% per year. Conservative estimates predict that as consumers, we will spend about six billion dollars in this emerging industry in 1983 [11]. It is difficult to imagine the magnitude of this. We have already come to an appreciation of the impact in our daily lives of the large commercial computers of the 1960s and 1970s. If all of those computers were to be turned off, the economy of the Western world would instantly grind to a halt. Telephones wouldn't work, power systems would shut down, and companies couldn't keep track of their business. Now, suddenly most small businesses and many families can afford a microcomputer with information processing power which far exceeds the typical commercial computer of the 1960s. And, the trend continues, with ever faster and more capable computers being announced in the market place each year.

Where and how is all of this computing power going to be used? Computer professionals have long felt that the advancement of technology and the availability of new computing hardware have always remained a few steps ahead of their ability to make full use of the power available. What, then, can be done with all the computer equipment now flowing off thousands of assembly lines?

One thing is clear, there are not enough computer experts around to make use of all the equipment. Beginning with hand-held calculators, microcomputers have already found their way into most every area of human endeavor. The need to understand computers is now required to some extent for the majority of professional categories with new areas of computer application being announced almost daily. All this has created the need for wide-spread "computer literacy" among professional persons. As a necessity, the new computers are being made easier to use by a larger variety of people, and, more and more people are computer literate enough to use them. Computer literacy is becoming a basic requirement for a high school education.

As a matter of basic survival, business executives and institutional administrators are making personal computers a basic tool of their trade [1,13]. The situation is rapidly approaching the point where, in most companies, the total information processing power setting around on desk tops will far exceed the power of the large centralized computer. The Management Information System (MIS) directors in many organizatins are concerned about whether all of this computer power will be coordinated to work in the best interests of their institutions [14].

This brings us to the key theme of our discussion. We are on the crest of a wave which cannot be denied, however, if an assortment of hardware and software is allowed to proliferate in an organization without any coordination with the central information system, a number of possible problems may result. First, it is much more expensive to maintain an assortment of equipment then to focus on a particular model or small selection of models. But, more importantly, if the assorted microcomputer equipment does not have any direct communication with the central information system, there will likely be a myriad of information related problems caused by multiple uncoordinated copies of the same information, inability to share information from other departments, inability to do coordinated analysis of information from several different departments, etc.

The challenge faced by the MIS director is to design a philosophy of integrating microcomputers with the central information system in a form which can be supported by the other departments in the organization. Doing this is not a simple matter. It requires an understanding of computing service economics, hardware and software evolutionary trends, changing user trends, changing roles for computer service centers, and political skill in gaining organizational support.

The next six sections are dedicated to discussing these factors. The remainder of the paper then outlines the philosophical approach which is being implemented at Linn-Benton Community College in Albany, Oregon.

Section 2

Computer Applications - Moving Closer to the User

Computer professionals are extremely fond of three character acronyms. So it is that the historical evolution of computer application software can be described as a movement from EDP to MIS and from MIS to DDS.

EDP stands for electronic data processing which was the starting point for the computer era in the 1960s and early 1970s. The emphasis was on automating clerical processes to reduce costs and to produce more timely results. In the mid 1970s, the emphasis began to shift to management information systems (MIS). The MIS concept hinged on digesting and structuring raw data into information reports which provided managers with timely insight into the performance of their operations.

We are now growing into a new era of computer applications which has been named decision support systems (DSS). This concept refers to applications which are designed to support the construction of information based decision models [4].

Understanding the DSS concept is one of the keys necessary for a successful marriage of microcomputers with a centralized MIS. The concept begins with user friendly application software with interactive user supportive input and convenient output with graphics capability. But, to be effective, it must extend beyond the software to include technical support and assistance to the decision maker in the development and refinement of decision making models [6].

The goal of DSS is to free the decision maker from the tedium of manual methods and from dependence on clerical and secretarial support. In general, it is designed to eliminate the time-consuming overhead that has traditionally limited the decision makers analytical capabilities [15].

Whereas EDP and MIS support primarily the lower and middle levels of an organization, DDS is designed to support the top levels of an organization. Executives at this level generally have some options. If the MIS director does not provide the required services, the executive will exercise other options such as stand alone microcomputers. Timeliness is therefore a key factor. The MIS director must come forth promptly with an effective plan for DSS or forever lose the chance.

Section 3

Recent Computer Hardware Concepts - Getting in Touch with the User

With the evolution of micro-electronics, the economics of computing hardware has changed drastically. Today's computers cost only a small fraction of what equivalent equipment would have cost ten years ago. This development has had a major impact on the computer to user interface.

Computer systems of the 1960s and early 1970s were designed to interface with users in ways which optimized the use of the computer but which gave only secondary importance to the convenience of the user. The systems were based on the concept of batch processing. Input for a computer run was usually prepared in punched card form. The card deck for many jobs from many different users was combined in a computer run batch. The batch was then processed on the computer usually without any user involvement. The output of all the jobs in the batch was then again processed as a batch on some printing system. The batch of output was then torn apart and distributed to users. The elapsed time from submission of input for card punching to the return of the printed output was typically several hours. The smallest error in the process, of course, would cause the run to be wasted, and the input had to be corrected and submitted again. At best, this was a frustrating and painfully slow process.

As computers became increasingly more cost effective during the 1970s, systems began to be designed to more directly and conveniently interface with the user. This gave rise to "on-line" systems with users communicating directly with a shared central computer through some type of communications terminal. In these systems, users began to get immediate response to the instructions which were sent to the computer. Errors could be corrected and tested again immediately. Obviously, users found the new systems much more desirable than the older batch processing systems, however, the costs were high. Sharing a large computer simultaneously among many users required extensive communications equipment and very complex and high overhead operating systems. The stage was set for ways to provide equivalent computer access at reduced cost.

In the 1970s and early 1980s, the emergence of minicomputers and microcomputers provided a basis for greatly reducing the cost of computing. The traditional large central computers were renamed "mainframes" to distinguish them from their smaller new competitors. But, the solution was not as simple as to replace an expensive computer with a cheaper one. For, in many uses, there was still the need to have large numbers of geographically dispersed users sharing a large information data base. This had been possible with the large mainframe system, however, the smaller minicomputers and microcomputers did not have the necessary data input and output capabilities to stand in place of a mainframe.

These circumstances led to new types of computer system architecture designed to merge the cost effective computing power of the smaller computers with the data handling capabilities of a mainframe computer. The result was the concept of distributed data processing (DDP). In DDP systems, smaller computers are located close to the users to do as much local computing as possible. The smaller systems have communications links with each other and/or with large mainframe systems to provide the necessary level of data sharing and coordination. The users assume a share of the responsibility for the operation and management of the systems in their areas. This new measure of user independence and self-reliance is being well received by users.

An effective approach to an organization's MIS must take these trends into account, thus, DDP concepts are a necessary and natural vehicle for integrating microcomputers into a large information system [9]. A necessary adjunct to DDP is defining an appropriate new role for the mainframe [12].

Section 4

User Relations - User Support is the Key

With all the changes in the architecture and philosophy of computer applications and computer hardware, it is not surprising that the relationships between a computer service center and the center's users are also changing dramatically. The rise of computer literacy and the widespread availability of personal microcomputers has made the typical user much more knowledgeable than just a couple of years ago. There are now more precisely defined ideas of user needs and expectations to which the computer service center must measure up. No longer does the user feel totally dependent on the computer experts. If the service center doesn't perform as desired and when desired, alternatives are available. The fact that some of those alternatives may not be in the best long range interests of the organization won't necessarily prevent them from happening. The situation is further sensitized by the fact that more and more top executive officers are becoming direct users [7]. Clearly, the MIS director must acquire and maintain broadly based user support in order to translate the microcomputer revolution into the best advantage of the whole organization. This can be done if some key concepts are recognized regarding the emerging relationships between the user and the information computer services center [2].

The old concepts of the process for developing a new computer application began with the user, supported by a system analyst, preparing a formal statement of "user requirements". After approval, the requirements statement document would be handed to a programmer for implementation. Weeks, months, or years later, the program would be made available to the user. This process was seldom totally satisfactory for a couple of reasons. First, few users were technical enough to precisely and accurately state their requirements. Secondly, because the user was not directly responsible for the implementation process, it was possible, or even likely, that the developed program did not fully meet the user's original expectations.

The new emphasis is on user responsibility and involvement with leadership and support from the "experts". This involves the use of general purpose applications like word processing, electronic spreadsheets, query languages, report writers, decision support systems, graphics, etc. It also involves more specialized application nucleus packages which can be easily extended and modified to meet individual user needs. Support from the computer service center is essential; however, increasingly the responsibility will be placed on the user to make use of the array of general purpose computer services available to effectively meet departmental needs.

All of this demands a flexible attitude from the MIS/Computer Services director. For, the focus is switching from the means to the end result. No longer can the director afford to fixate on the beauties of hardware and software. Instead user needs must be center stage [5].

Finally, it must be recognized that the user is claiming a new proprietary interest in computer applications. This can be guided into a constructive involvement in the implementation of distributed data processing systems designed to work in harmony with the central mainframe computer. If this opportunity is missed, an equivalent amount of energy will likely go into proliferation of stand alone microcomputers and rejection of mainframes and any form of centralized computing [10].

Section 5

The Director - Struggling to Keep Pace

Recognizing the nature of all the trends and changes is important, but, it's only the first step. The key challenge to the MIS/Computer Services director is to reconfigure services rapidly enough and on-target enough to meet the needs of the hour. This is no easy task. It feels somewhat like trying to construct a building on top of a foundation that is constantly shifting. One must struggle to look far enough ahead so that new services don't become obsolete by the time they are implemented. All the while, the basic factors upon which the new systems are built are shifting with new developments in technology.

To begin, the whole concept of strategic long range computer service planning is relatively new to many computer professionals. Back in the days when MIS directors were called EDP directors, most planning consisted only of projecting volumes - number of reels of tape, boxes of paper, bytes of storage, terminals, etc. Almost all strategic planning was done by equipment vendors. EDP professionals simply reacted to what the vendors did. But, the situation has drastically changed. Today, effective computer services management requires an assessment of trends in technology, constant searching for and analysis of alternatives, and an insightfully constructed long range plan to address the needs of the organization. Any center without effective strategic planning will soon be left wandering hopelessly out of touch with the users.

Next, the driving forces behind decisions are becoming entirely different. In earlier times, decisions to develop computer applications were based primarily on economic considerations such as eliminating expensive manual operations, or reducing costs through operational efficiencies. Such considerations are now sometimes secondary. The new priority is on promoting the organization's information resource. It is difficult or impossible, in most circumstances, to place an objective value on information, hence, developments done in the name of information cannot be precisely cost justified. Instead, only a subjective assessment of cost versus benefits can be made. Information is being viewed as a new form of wealth necessary to the well being of the organization, and decisions are made on that basis [8]. A related but somewhat different rationale for new systems is the new area of decision support. How can we calculate the value of computer assisted decision making? Again, only a subjective evaluation can be made.

Finally, at a time when so much change needs to be made, most MIS/Computer service operations are experiencing the leanest budgets in many years.

All of these factors combine to add confusion and to invalidate traditional approaches to solving problems and making decisions precisely at the time when creative decisions must be made and bold actions must be taken. Truly, the microcomputer is turning the computer professional's world upside down. But, the opportunities are even greater than the challenges. The computer professional who can tame and coordinate all of the new equipment that is daily flowing into organizations everywhere, will be harnessing a force with enormous potential benefits.

Section 6

Planning Goals - Setting our Sights

The primary purpose of the discussion up to this point has been to provide insight into the motivation for the planning goals to be listed in this section. The relative importance of the goals may vary from one situation to another, but, it is likely that they are all universally important in some significant measure. With this in mind, we will simply list them without any attempt to place them in a priority order.

Goal I. Foster user involvement in and commitment to MIS, DSS, and other computer services. This should include giving users responsibility for operation of and budgeting for local computing equipment.

Goal II. Provide word processing, electronic spreadsheet, decision support packages, graphics, and related applications as close as possible to the user on local equipment.

Goal III. Provide user training and support necessary to help users meet local responsibilities and make optimal use of the available services.

Goal IV. Control the proliferation of stand alone equipment, and provide means for coordinating all equipment into a master organizational plan.

Goal V. Incorporate the cost effective computing power of microcomputers into the central MIS through distributed data processing techniques.

Goal VI. Provide centralized coordination and control of the information resource. Make information directly accessible on-line to the users needing it.

Goal VII. Define an appropriate role for mainframe computers based on their unique ability to manage large volumes of shared information.

The above goals outline an approach to designing an effective plan for using today's computer technology, microcomputers in particular, to the best advantage of the total organization. In any particular situation, we would need to add a list of specific organizational objectives before we are ready to develop a computer services plan. In the next section we will review how this was done at LBCC.

Section 7

A Plan - Putting It All Together At LBCC

The author joined Linn-Benton Community College in July of 1981. There were already some high expectations at the college regarding what was needed in the area of computer services. It was relatively easy to translate these into some specific objectives for computer application development. Designing a plan to implement the objectives in times of severe economic constraints was a much greater challenge. We'll begin with a discussion of the objectives.

First, there was an objective to automate college business functions. This involved primarily the business office and the admissions and registration office. There were two aspects to the motivation for this. One was to present an efficient service oriented interface with our customers - the students. These changes are apparent to students in the form of much shorter waiting lines during registration and prompt processing of refunds and other business transactions. The other aspect was to provide timely operational information such as budget account balances, enrollment figures, etc.

The approach was to make the new services available through on-line computer terminals having direct access to the centralized college data base. These services replaced a combination of manual procedures and batch oriented EDP.

Next, there was an objective to develop a state-of-the-art MIS. Two of the key motivations for this were to automate external reporting requirements and to provide information for timely management decisions particularly in the area of budget.

Finally, there was an objective to make computer center resources more accessible to instructional support functions and student use.

Preliminary analysis indicated that in the neighborhood of 100 computer terminals would be required to provide the necessary level of system accessibility. This involved transitioning from a primarily batch oriented system with a handful of on-line terminals. It would require lots of new software and more computing power. This at a time when the college felt that it was already spending as much as it could affort on computer services. Where was all the new hardware and software going to come from? With this frame of reference, it was our hope that new technology and microcomputers in particular might provide part of the answer.

The stage is now set for describing the LBCC Computer Services Network architecture. It was designed to meet a specific set of needs and is not intended as a prescription for other institutions. However, it does illustrate some principles that could be applicable to a variety of circumstances.

Section 8

LBCC/NET - A Delivery System

The fundamental purpose of a computer service network is to deliver information and computing power close to the user where and when needed [3]. This is accomplished by data communication links connecting the terminals and computers in the network. The links are shown in figure 1 as lines connecting the network components. The components themselves are shown as rectangles. There are three basic kinds of them. The mainframe computer is labeled MAIN. Multiuser microcomputer network components are labeled μNET. User terminals, printers, or personal microcomputers are classified as user devices and labeled U for user device.

The network architecture is basically wheel shaped with MAIN at the center and ten μNETs around the rim. Each μNET has a lateral link attached to its neighbor. This allows communication around the rim without mainframe involvement. Each uNET also communicates directly with the mainframe. A μNET supports up to twelve user devices. There are currently ten μNETs in the network representing a potential of about 120 user devices.

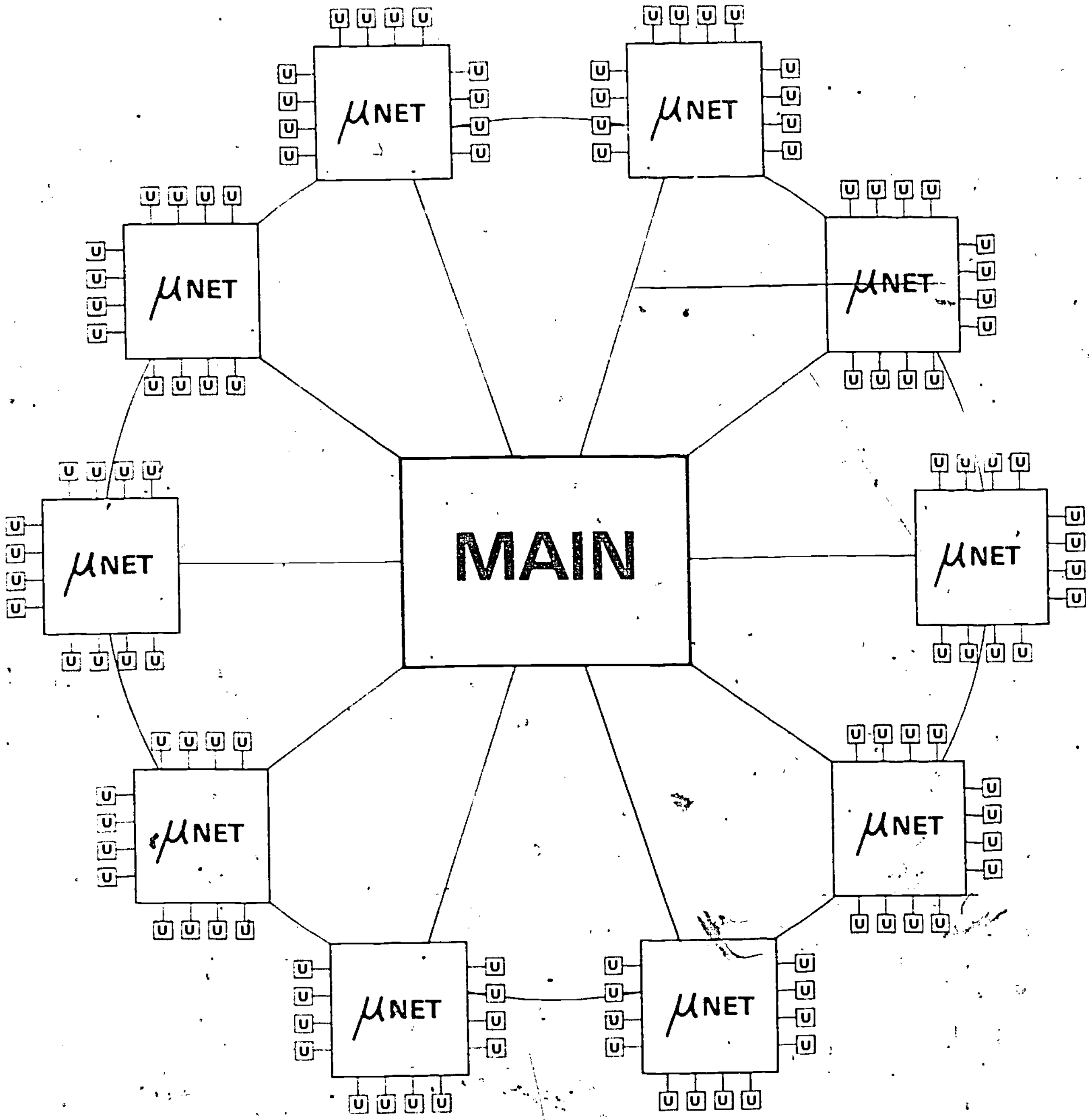


FIGURE 1

LBCC/NET ARCHITECTURE

μNET = NETWORK MICROCOMPUTER

U = USER DEVICE

Progress to Date - LBCC/NET's Report Card

In terms of quantity of equipment, the network has already completed its major period of rapid growth. Excluding the mainframe system, the network component list is as follows:

10 - Network microcomputers
90 - Display terminals
14 - Local Dot Matrix Printers
5 - Letter Quality Printers
4 - 300 Line/Minute Printers
4 - 1200 Baud Dialup Modems

In terms of planned communications capabilities, the network is only about 50% complete, however, it is providing all of its planned services in substantial measures. Thus, a final assessment cannot yet be made, but we can track our progress to date on the goals and objectives which have been defined.

The first goal relates to user involvement. At LBCC, the users were given full responsibility for planning their portions of the network with support from the data processing staff. Users budgeted for portions of the equipment. They are also responsible for its operation, again, with the necessary support from the data processing staff. It is difficult to describe the profound effect which all of this has had on the users' perceptions of computer services, however, it will suffice to say that the goal of user involvement has been achieved.

The second goal involves providing general purpose applications which are especially cost effective in a microcomputer environment such as word processing, electronic spreadsheet, decision support, etc. A number of these have already been made available on the network microcomputers for use by the connected user devices. So far, word processing is getting most of the attention at LBCC. For some time, the college has been recognizing needs in the word processing area which have been going unfulfilled because of budgetary limitations. This need is now being met as a low cost byproduct of LBCC/NET capabilities.

The third goal concerns providing the training necessary to enable users to manage and make best use of their computer services. At LBCC, this is being accomplished with the concept of an "area computer specialist." These are persons from the user areas appointed to take a lead role in working with the data processing staff to acquire a level of expertise regarding the computer applications in their areas. And, in turn to take the lead in training other persons from their areas. With the help of area specialists, the training load on the data processing staff becomes manageable. This is proving to be an effective approach to the training requirement.

The fourth goal aims at controlling the proliferation of stand alone microcomputer equipment without any master plan. There are two aspects in the LBCC approach to this issue. First, there is an institution wide long range strategic plan for computer services. Secondly, all computer related purchase orders are reviewed by the data processing staff for consistency with the plan. This policy has been instrumental in helping the college move toward an integrated institutional approach to computer services.

The fifth goal attempts to take advantage of the low cost computing power of microcomputers wherever possible in place of mainframe computing. At LBCC, the long range plan calls for shifting as much load as possible from the mainframe to the network microcomputers. Eventually, the mainframe will be primarily used for supporting the network access to the centralized data base. Already, the existence of LBCC/NET has obviated the need for an expensive mainframe upgrade.

The sixth goal strives to coordinate the organizational information resource in order to maximum its benefits and protect its integrity. LBCC/NET makes this possible by making the centralized college data base accessible to all user devices including personal microcomputers. There is therefore no need to keep separate fragmented and uncoordinated versions of college information. This is of crucial importance to an effective MIS.

The seventh and final goal assigns a new personality to the mainframe computer. The basic principle, based on cost consideration, is to use the mainframe only for functions which require its more powerful information handling capabilities. This principle is a key part of LBCC/NET evolution plan.

To summarize, although LBCC/NET still has a considerable way to evolve to reach all its planned functional capabilities, measurable results are already being realized on all the fundamental computer services goals we have identified. In terms of the specific LBCC objectives of automated on-line business functions and management information, LBCC/NET has likewise catalyzed substantial progress. Many college business functions including student registration are now done through on-line terminals. On-line up to the minute budget account information is currently being installed to assist managers with expenditure control. Management information needs are being met in a more timely and effective manner than before LBCC/NET. The key to all of this has been acceptance of the microcomputer as an important partner in the college's computer services in general and in its information system in particular.

Section 10

Prospectives - What Happens Next?

We can predict that, for the most part, LBCC/NET will unfold according to the long range plan, but there is a far more important aspect involving user anticipation regarding the future of the network. This is related to a sense of participation in the unfolding of computer technology. There is the feeling that as new devices are developed the users will be able to obtain and take advantage of them. Part of this is due to the excitement over what has been done in LBCC/NET, the other part is due to price considerations. Microcomputer oriented products often sell in the hundreds of dollars where similar products in a mainframe environment often cost many thousands of dollars. Thus the optimism that we will be able to obtain and employ the latest technology.

To conclude, the microcomputer revolution is bringing about a most positive transformation in user attitudes toward computing. Users are displaying a "can do" expection regarding future network capabilities. In some cases, this attitude has replaced an expectation of unsatisfactory results and frustration based on years of experience with a centralized mainframe approach to computing.

References

1. James E. Braham, "You Don't Have a Computer?" Modern office Procedures, August, 1982, p. 84.

2. Robert Danielenko, "Keeping Users Happy," Computer Decisions, April, 1983, p. 82.

3. Susan Foster, "Networking Micros...Sharing Resources," Computer Decisions, April, 1983, p. 100.

4. David Kull, "The Dawn of IRM," Computer Decisions, October, 1982, p. 94.

5. David Kull, "Information Centers Help Users Put It All together," Computer Decisions, Febebruary, 1983, p.70.

6. Martin Lasden, "Computer-Aided Decision-Making," Computer Decisions, November, 1982, p. 156.

7. Martin Lasden, "Make Room for Executive Workstations," Computer Decisions, December, 1982, p. 116.

8. James Martin, "Assessing the Destiny of Information-Part 2," Modern Office Procedures, April, 1983, p. 55.

9. Mary Miles, "Gearing Up for DDP," Computer Decisions, April, 1983, p. 128.

10. Frederick W. Miller, "Are Mainframes Crumbling from Outside Pressures?" Inforsystems, July, 1982, p. 80.

11. MOP Staff, "Micros Herald DP Changes," Modern Office Procedures, January, 1983, p. 48.

12. Wayne L. Rhodes Jr., "Mainframes - Getting a New Act Together," Infosystems, September, 1983, p. 30.

13. David R. Roman, "Executives Who Love their Personal Computers," Computer Decisions, January, 1983, p. 142.

14. Steve Sibbins, "Microcomputers: Friend or Foe of the MIS Manager," Infosystems, October, 1982, p. 44.

15. Carol Tomme Thiel, "DSS Means Computer-Aided Management," Infosystems, March, 1983, p. 38.

DEVELOPING AND DIS TING MICROCOMPUTER SOFTWARE

Kenneth E. Brumbaugh

e availability of high quality microcomputer courseware for educators wil improved as both consumers and producers become more computer literate t tter understanding the process of microcomputer courseware development ar e varying perceptions on its distribution. This paper is aimed at sisting that literacy goal.

General Paper

Comments

troduction. As the second generation - hardware, courseware, and users olves in the educational microcomputing arena, so too do the problems cing users of computers in schools. Problems often mentioned are: not ough microcomputers, unavailability of quality courseware, insufficient cess to in-service training, and lack of K-12 curriculum planning. As re individuals gain more experience in the use of classroom computing, it uld seem that solutions to some or all of the above problems should be rthcoming.

e problem of not having sufficient computer stations in a classroom will eatly diminish in this decade. Colleges, universities, and educational rvice organizations should begin to meet the in-service training needs. atewide efforts such as those in Alaska, California, Indiana, Minnesota, w York, and elsewhere will begin to address the issue of curriculum anning related to the use of computers in instruction. Even being able t tain good general purpose courseware should result, particularly if ucators have a better understanding for how courseware is produced and stributed.

rpose. The purpose of this paper is to present information on the topic developing and distributing microcomputer courseware. Development of urseware will be presented from a procedural standpoint, while stribution of courseware will be presented from a perceptual standpoint. e topic of courseware evaluation will not be addressed.

ckground. When one thinks of history a long period of time comes to nd. In the case of developing and distributing microcomputer courseware r education the period of history is limited to five years. It was proximately five years ago that Milliken Publishers, St. Louis, Missouri gan to produce what became the first successful commercial microcomputer urseware product for education. Now more than 150 companies are known fc eir line of educational microcomputer courseware products.

Early problems for the education courseware industry included, but were not limited to, individual as compared to team efforts, singular and restricted modes of delivery as compared to multi-modal applications, and lack of any method for 'outside' distribution. Individuals who became good at programming started to produce drill-oriented, textual computer programs and touted such as education courseware. The new available features of computers, such as graphics, animation, and sound were not being fully utilized. When a good program was produced, it was unlikely that it would ever get wide dissemination. Most of these problems have gone away as experienced courseware developers have evolved, and as a critical mass of software resulted in distribution channels being established.

Development of Courseware

Definitions . Throughout this paper the term courseware is meant to be a combination of the microcomputer program(s) and the associated written user (teacher, parent, or student) support material necessary for successful and worthwhile use of the product. The support documentation, although sometimes only used for a few occasions, raises the probability of proper use and lasting value associated with the use of computers in educational settings. It can also play a valuable role in determining if such an application should even be used in the particular educational setting.

Since courseware is the most popular term associated with the use of computer applications in instruction I shall use it in this paper. Lessonware, or modules, might be better terms to use in that most 'courseware' currently is stand-alone and supplemental in nature, and far from being a 'course.' I suspect that we will see more courseware in the future as individual applications are tied together and as teacher management aids are added to software packages.

People . In all stages of courseware development a team of individuals will produce more usable and more universal materials than an individual working alone, regardless of the time devoted to task. Development team members should have expertise and experience in the areas listed in Exhibit 1: Courseware Development Team Experience Requirements - courseware design and development, microcomputer programming, subject matter/discipline, teaching, and personnel management. If each team member had experience in all categories it would be ideal, though very unlikely. The number of individuals, who have the multiple experience base of using the computer as an instructional tool and who have curriculum development experience, is growing.

EXHIBIT 1

COURSEWARE DEVELOPMENT TEAM EXPERIENCE REQUIREMENTS

Courseware Design & Development
Microcomputer Programming
Subject Matter/Discipline
Teaching
Personnel Management

Proper management of the development team is a critical requirement for success. This manager must be able to correctly and carefully assign tasks, establish deadlines and schedules, and coordinate the combined efforts of all team members. Both individual and group human relation skills are necessary at various steps of the development process. Assisting a designer layout a sample screen requires individual communication, where conducting a structured walk-thru of the programming revisions to be made necessitates group communication skills. The manager must decide who will be responsible for each step in the process and which team members to include at each step. Perhaps the hardest management task is assignment of completion dates for the numerous tasks throughout the entire development process. Courseware team development is similar to conventional manufacturing production; time will be wasted and costs will escalate if proper coordination and scheduling are not done. An advantage of the team concept is improved productivity since work can be assigned to the most qualified person, thus reducing the time devoted to task.

Process . Just as there are multiple ways to get between two points, there are many ways to develop instructional microcomputer courseware. If someone else is to follow your path, or others need to make the same journey, then it becomes imperative to chart the course from beginning to end. Having a well documented courseware development procedure will not only facilitate getting the initial product created, but will greatly simplify any later revisions.

Once a decison is made to create a microcomputer courseware product the development process could include the steps as listed in Exhibit 2: Courseware Development Steps - project initiation, design, development, programming, testing, modification, review, refinement, polishing, and production. Each of these steps could have many substeps included in them. Various development team members would assume the responsibility for moving the product thru each step, or substep.

EXHIBIT 2

COURSEWARE DEVELOPMENT STEPS

Initiation
Design
Development
Programming
Testing
Modification
Review
Refinement
Polishing
Production

Identifying and detailing the various substeps in the development process is the key to success. Each step in the development process would have substeps which would result in different individuals producing a specified result at a certain time. Consider the initiation step, some of its substeps could be: establishment of timelines and budgets, assignment of team members and leaders, and selection of an advisory group. A second example of substeps, in the programming area, could be: assignment of programmers and lead programmer; preparation and review of screen layouts; preparation, review (structured walk-thru), and modification of program design; program coding and debugging; program review (structured walk-thru) and testing; revisions; and program documentation. The entire process could have more than fifty (50) sub steps.

Structured walk-thrus are group sessions in which the individual responsible for that task presents to other team members the order and rationale for each included phase of either the planned, or completed work. Group review sessions of this nature will insure that product meets a larger universe of needs, has less errors, and in general gets done on time and in budget. Structured walk-thrus may be resisted by development teams at first, but after tried, most will look forward to getting the assistance and support of to theircolleagues.

<u>Products</u> . Development of the microcomputer programs and related user support documentation is not completed until the decision on packaging is made. How many programs will be combined on one diskette? Will there be an user booklet for each program? What media will be used to deliver the product? These are some of the questions that must be answered during the development process so that intended dissemination will ocurr. After a period of months, and sometimes years, the product is released from the development team and is ready for production. Production usually means printing of the written materials and copying of the associated computer medium.

Distribution of Courseware

<u>Perceptions</u> . Too often educators do not see the various views on topics which relate their teaching situations. In the case of educational microcomputer courseware distribution a variety of viewpoints exist, including those listed in Exhibit 3: Courseware Distribution Perceptions - end-user, developer, distributor, and retailers. For schools to obtain the desired quantity of high quality courseware all of the involved parties must be satisified. Perhaps the first step is to improve the awareness of each others' perceptions.

EXHIBIT 3

COURSEWARE DISTRIBUTION PERCEPTIONS

User
Developer
Distributor
Retailer

The end user of a courseware product has high expectations, perhaps too high, for the price paid. The price of educational computing courseware products may range from $25.00 to several hundred dollars. A user has every right to expect some type of review policy and to be able to get, or obtain, an inexpensive backup copy of the medium. With the exception of licensing arrangements for local computer networks, no user should expect to buy one copy of a product and then be able to use it with multiple computing stations. Users should view and honor the rights of courseware authors in the same manner as textbook authors. The fact that computers can copy a product, or copy machines can copy a book, does not make it legal, or morally right to do. Proper planning and budgeting will assist the end user in obtaining adequate quantities of approriate courseware.

The developer of a courseware product has created something that has value for others and is confronted with the problem of finding a method to distribute it. A considerable investment of time and resources has been expended and thus a return on that investment is desirable and reasonable. That person or organization has created something that also has value for others and is confronted with the problem of finding a method to distribute it. Some of the 'microcomputer' related questions facing the developer include: which machine to use, which model, which operating system, which language, and which medium. Using a compiler type language is a partial solution to some of the above problems in that it is easier to switch source code versions when necessary. Getting advance product information from the microcomputer manufacturers can soften the 'new' model transition problem.

Distribution of microcomputer courseware has become a big business and professional distributors are now the principle movers of courseware products. Many of the business procedures used in the distribution of records and films are being used to handle microcomputer courseware products. Product awareness is perhaps the biggest hurdle for distributors and many dollars are spent on advertising. Most distributors 'protect' their software by some type of security system because their profit is based entirely on quantity of sales. Prompting filling orders means that distributors must keep adequate product inventories, or be able to produce the items in short time. The normal billing and accounting problems exist and are compounded by late, or in some cases no payment. I suspect that educational users who desire a high quantity of available product titles will have to look for distributors who specialize in serving them. Many of the audio-visual mail order supply houses are now handling large amounts of educational computing courseware. The major textbook publishers are beginning to offer courseware products, particularly those that relate to their texts.

The retailer, who normally handles microcomputer hardware and software, is interested in sales and service. Since the profit margin on many hardware sales is small, the retailer has two goals related to software sales. One is to have the customer return to buy software, and the other is to bundle software sales in with the original hardware purchase so as to make additional profit. A growing problem for retailers is 'shelf space', particularly as more and more product becomes available. Retailers for the most part do not carry very many software product titles for educators (courseware.) The reason is lower courseware prices, as compared to the higher cost business titles, and low volume of sales. Educators would be well advised when buying courseware from retailers to shop around and to purchase software when obtaining hardware.

Service Centers. State, regional, and local non-profit service centers are becoming one of the best sources for educators to obtain courseware. Such agencies are either doing collective purchasing and resale to schools, or are obtaining licensing arrangements for schools with major developers, distributors, or dealers. In essence these agencies are becoming special distributors, who, because of their service committments to schools, can provide the products at reduced prices. The best known example is the Minnesota Educational Computing Consortium.

Minnesota Educational Computing Consortium

General Information

History. The Minnesota Educational Computing Consortium (MECC) was created 10 years ago for the purpose of serving educational computing needs in the following Minnesota members: the Department of Education (representing the 437 school districts), the 18 Community Colleges, the 7 State Universities, and the University of Minnesota. MECC's staff, which exceeds 100 individuals, work primarily in the instructional computing area, although some work is done in administrative data processing for the school districts. In recent years MECC has obtained a non-Minnesota clientele consisting of 100+ members located throughout the world. The major accomplishments of MECC include the development and distribution of microcomputer programs, the establishment and administration of volume purchase equipment contracts, the provision of all-purpose educational computer in-service training, and the development of administrative computing software packages.

Microcomputer Courseware Activites

Development. Perhaps MECC is best known for its available library of 500+ APPLE II microcomputer programs. These programs are packaged together with the necessary user support material to comprise 90 courseware products. The MECC development philosophy has been to produce good computer courseware for all educators at all levels of education. MECC adds several courseware products per month to that base of microcomputer materials. Development steps similar to those outlined earlier are used by MECC to produce these products. The average development completion time per product is 12 months.

In order to serve the growing diverse educational computing user base MECC has entered into a series of agreements where by MECC has, or will, convert substantial portions of our APPLE II courseware products to run on other popular microcomputers. The Atari conversion effort is completed and has yielded 18 products (100 programs) primarily for use in the elementary school area. MECC recently has decided to convert programs for use on: the IBM Personal Computer (100 programs), the Radio Shack computers TRS-80 Color (20 programs) and the Model III/IV (20 programs), and the Commodore 64 microcomputer (100 programs.) Results of these last conversion efforts will be available in 1984.

Distribution. MECC courseware products are distributed through unit sales by MECC and by several commercial companies, and by a special member licensing arrangement called an institutional membership. Within Minnesota MECC courseware has been provided either at no cost, or at the cost of production and handling, saving Minnesota educators approximately $4,000,000. per year, should they have purchased the same materials at standard retail prices. An estimated 1,000,000 copies of MECC diskettes have been made available to computing users throughout the world. MECC's unit sales exceed 200,000 products per year, which does not include the products made available to our institutional members. Thirty eight of the United States now have statewide, or partial, memberships in MECC, enabling them to obtain and distribute MECC courseware products at reduced prices.

Other Activities

In-service Training. MECC offers a variety of training activities ranging from short sessions to complete courses. Nine hundred (900) general computing presentations are given each year to Minnesota educators. One hundred fifty (150) workshops are offered each year. The topics for these workshops, which are available to non-Minnesota educators, are listed in Exhibit 4: MECC In-Service Training Topics and include: computing's fit in schools; locating and evaluating equipment, courseware, and services; operation of computers; use of courseware; and computer programming for educators. MECC is attempting to concentrate more effort on training related to school planning and preparation for use of the computer in instruction. Emphasis is also being placed on teaching the programming languages LOGO and PASCAL. When MECC staff are satisified that a MECC developed in-service training guide has received sufficient classroom use, it is released as a MECC training product and is available for in-service trainers to obtain. Ten (10) training guides are now available for purchase from MECC.

EXHIBIT 4

MECC IN-SERVICE TRAINING TOPICS

Computing's Fit in Schools
Locating/Evaluating Computing Materials
Operation of Computers
Use of Courseware
Computer Programming

Administrative Microcomputer Aids. Although the majority of MECC's experience in administrative data processing for schools has been with large main-frame computers (Burroughs), a significant effort has recently been mounted to begin producing microcomputer-based products for school administrators. The first product released was an all-purpose data base package for use with small data bases, such as address lists for mailing labels. Both a personnel/payroll and a finance package will be released during the summer of 1983 with numerous other products planned for release in 1984.

Summary

The goal of this paper was to review the topics of courseware development and distribution by presenting information related to the people, processes, and products involved in microcomputer courseware development, and by sharing different perceptions of those involved in the distribution of microcomputer courseware products. Finally some information on MECC, the largest producer and distributor of microcomputer courseware for educators was presented.

The future will bring change, that is certain! Educators who understand the development and distribution of courseware will be better prepared to make it available for their use, regardless of how it developed, or distributed to them. The number of educators getting involved in either, or both, of these growing fields is large; perhaps you will be one of them.

References

Anderson, Ronald E., Hunter, Beverly, and Seidel, Robert J., Computer Literacy . New York: Academic Press, 1982.

Coburn, Peter, Kelman, Peter, Roberts, Nancy, Snyther, Thomas F.F., Watt, Daniel H., and Weiner, Cheryl, Practical Guide to Computers in Education . California: Addision-Wesley Publishing Company, 1982.

Goodson, Bobby and Lathrop, Ann, Courseware In The Classroom: Selecting, Organizing, and Using Educational Software . California: Addisoin-Wesley Publishing Company, 1983.

Minnesota Educational Computing Consortium, Designing Instructional Computing Materials . Minnesota: MECC, 1981.

Moursund, Dave, School Administrator's Guide to Instructional Use of Computers . Oregon: ICCE

Office of Technology Assessment, Informational Technology and Its Impact on American Education , Congress of the United States, Washington, D.C., 1982.

Texas Education Agency, Guide for Selecting a Computer-Based Instructional System . Texas: TEA

USING THE COMPUTER AS WRITING TEACHER: THE HEART OF THE GREAT DEBATES

Andrea W. Herrmann

The great debate which has been taking place in the world of writing instruction for some time mirrors the emerging debate concerning the implementation of computers in education. In the Great Writing Debate the central issue concerns whether writing can be taught through a mechanics-usage approach--grammar, punctuation, spelling--particularly via the manipulation of words and sentences, such as in workbook exercises, or whether writing instruction needs to rely primarily on the creation of written texts by the student, with mechanics and usage a by-product of the process, dealt with in relationship to the student's writing on an ad hoc basis. In the Great Computer Debate a war rages between the adherents of pre-programmed instruction--computer-assisted (CAI) and computer-managed (CMI) of the drill and practice and tutorial sort--versus those who advocate using the computer in more holistic ways with the student the creator or programmer of the activities.

The Great Writing Debate and the Great Computer Debate share a common philosophical foundation. The traditional grammar approach, like computer drill and practice, supports the underlying notion that isolating activities into classes makes them easier for students to understand, to learn, and to apply to the larger learning tasks. Opposition to these activities, however, suggests that they are mechanical, done rotely by students, and that information using these techniques is either poorly learned, irrelevant to more holistic tasks, not capable of being applied to new situations or that the segmentalization of steps fails to take into account the range of complex skills needed in the larger processes.

Seymour Papert, author of the best selling book, Mindstorms: Children, Computers and Powerful Ideas, sees the division as one between the computer as "teaching instrument" and the computer as "writing instrument" and states that "this difference is not a matter of a small and technical choice between two teaching strategies. It reflects a fundamental difference in educational philosophies." [1]

In looking over the literature on computers and writing, the existence of this dichotomy is striking. Applications and research fall into one or the other category: the computer as a teaching instrument of the basic skills or the computer used in holistic ways as a writing tool.

Basically two kinds of criticisms are made of computer-assisted programs. First is criticism that could be leveled at textbooks, namely that the content of the material about the nature of writing and the writing process is of questionable value based on writing research. A corollary to this kind of criticism is that the computer is being used only as an expensive workbook; many programs essentially do nothing that couldn't be done as well on paper. Second are questions about the pedagogical approaches. They usually rest on the assumption that students should learn grammar before writing. They may assume that students who can put in the correct form of a verb, pick out the topic sentence in a paragraph or find the word that is incorrectly capitalized are learning how to write. In fact, while these are all useful editing skills, they do not help students acquire or improve their abilities in topic selection, focus, coherence, cohesion, the elaboration of ideas or any of the many other activities that involve the creation of written text.

The risk that confronts English teachers who turn to the typical CAI programs as a means of teaching writing is the same problem English teachers confront in the workbook orientation to teaching writing: the skills that are usually being taught are not writing but editing skills.

The belief that grammatical form should take precedence over meaning as the preferred way to create effective writing is greatly disputed by many writing theorists. James Collins, in "Speaking, Writing, and Teaching for Meaning," claims that students taught from this premise become effective in "error avoidance" and that their writing is brief, "vacuous and impersonal, polite and innocuous." [2] Anthony Petrosky deals with the issue of grammar in an article, "Grammar Instruction: What We Know." Based on a review of the literature, especially two carefully conducted longitudinal research studies carried out on the value of grammar instruction to writing improvement, he concludes that the study of grammar has no influence on the language growth of typical secondary students and that "there is no empirical evidence for the teaching of grammar for any purpose." [3]

A new model of the writing process has evolved as a result of the work by contemporary writing process theorists and researchers--Sondra Perl, Donald Murray, Donald Graves, Lucy Calkins, Janet Emig, Linda Flower and John Hayes, among others. They are exposing the fallacy that writing is a linear series of sequential steps proceeding from pre-writing to writing and then to revision. Methodologies and contexts for studies remain diverse and include the ethnographic case studies of unskilled college writers by Perl, the use of laboratory protocol analysis of writers speaking into tape recorders as they write of Flower and Hayes, and the studies of children in schools by Graves and Calkins. However, they're discovering phenomena, to a great extent interrelated, that create a new and presumably more accurate view of the real nature of writing.

Perl describes it as a recursive, back and forth shuttling process. [4] She talks about it as one of "retrospective" and "projective structuring." [5] Flower and Hayes say,

> the writing process, like any other creative process, is rarely straightforward or direct. A writer's conclusions, his main ideas, even his focus, are often the product of searching, trial and error, and inference. [6]

They also point out the potentially negative influences of the parts-to-whole approach in teaching writing.

> This process could easily be disrupted by focusing on form too early. Thus a product-based plan may thwart the dynamics of the normal generating process by placing unnecessarily rigid constraints in the early stages of the writing process. [7]

Ample evidence exists to question approaches to writing whose principal concern lies within subskills--such as grammar, usage or form--rather than meaning and which draw their assumptions from the idealized model of the linear conception of the writing process that no longer appears valid. Emig makes a strong case for writing as a "unique mode of learning" and shows it to be organic and functional, a way of making learning connective and selective. [8] One of the dangers of time spent in inneffective ways of teaching writing is noted by Petrosky in his final evaluation of the role of grammar in teaching writing: "The study of grammar, while serving no ascertainable purpose, also exists at the expense of proficiency in reading and writing." [9] There is a fear, as Emig claims, that

> unless the losses to learners of not writing are compellingly described and substantiated by experimental and speculative research, writing itself as a central academic process may not long endure. [10]

Yet there seems reason to be optimistic about the teaching of writing, the role of the computer in that process, and, perhaps, even reason to believe that the current interest in writing and computerized instruction may serve to create a new emphasis and new strategies in that art. Recent approaches in the area of computers and writing have attempted to shift the focus from the computer as a teaching instrument to one where the student takes a more active role and the computer becomes a writing tool.

Word processing is probably the most common way the computer is used holistically as a writing tool. One of the important questions concerning this approach is how the use of computers affects the composing process. Is writing done on computers different from, perhaps superior to, written work done using other tools?

The question is legitimate. The current electronic age has given us a new sensitivity to the differences between the word as sound and as print. [11] According to Walter Ong, the word has been transformed in three stages: oral, script and electronic. [12] Jack Goody and Ian Watt take an in-depth look at the idea that "writing established a different kind of relationship between the word and its referent, a relationship that is more general and more abstract, and less closely connected with the particularities of person, place and time, than obtains in oral communication." [13] Ong also argues that, "more than any other single invention, writing has transformed human consciousness," by establishing "context-free" language as opposed to the embedded nature of oral discourse. If one accepts their argument that the means of production of thought, oral vs. literate, affects the nature of thought produced, then one may conclude that the different means of producing literate thought--pencil, typewriter, or word processor--could exert significant influence on the nature and quality of the written product.

If so, how might these differences manifest themselves? Writing done on a computer could affect the number and quality of ideas; the correctness of grammar, usage, and spelling; the choice of vocabulary; the complexity of syntax; style; and many other aspects of writing. Unfortunately there is as yet very little research evidence to indicate whether or not such effects do take place.

Burns and Culp have experimented in a Freshman English setting with a program that attempts to break away from the drill and practice format and to encourage students in "the process of exploring a subject to discover ideas, arguments, or propositions--those features which one must know in order to write convincingly about a subject." [14] Their conclusions state that their program encouraged both growth in the number and the sophistication of ideas." [15] This research did not involve the writing of compositions, only the generation of ideas on a topic the students had selected for a research paper. It did not evaluate the number and sophistication of ideas actually used by the students in the eventual creation of their papers. No conclusions can be drawn, therefore, as to the effectiveness of the program to generate ideas in the actual composing process.

Two studies done on computer assisted programs to help children handle structural elements of the composing process were carried out by Earl Woodruff, Carl Bereiter, and Marlene Scardamalia. Unfortunately what appear to be faulty assumptions about the composition process and the subjects' lack of experience in typing and word processing resulted in the creation of ineffective programs on the one hand and inconclusive results on the other. The first study concluded that the program was deemed "not to actually have engaged the students in a higher-level consideration of the composition choices" but resulted in students taking a "what next?" approach to their planning. [16] The second study, which continually interrupted the students as they composed to ask them response-sensitive questions designed to "foster more carefully considered and more fully developed essays," resulted in work which received lower ratings. [17] Writers in the act of composing are bound to be disturbed, it would seem, by questions, no matter how well intentioned. The encounter with this new strategy, while long enough to show its ineffectiveness, was too short, even if it had been a pedagogically sound one, to show improved written work. The ability to master new strategies may be a much slower process than researchers realize.

Research into the possible effects of word processing on writers and writing is currently being carried out at the University of Minnesota. The project, a three year plan, is looking at the composing process of writers and the pedagogical implementation of word processing in the context of the classroom. Results, however, are not yet available.

Studies on the effects of word processing with computers on children's writing are currently being carried out at Teachers College, Columbia University, by Colette Daiute. Her preliminary evidence suggests that word processing improves the quantity of writing, the number of revisions, and the length of the manuscripts done by children. [18] These findings, while in themselves insufficient evidence to conclude that the computer has affected the quality of the end product, lend support to that possibility. Studies on the composing process, especially revision, highlight the relationship between revision strategies and the quality of writing. It is probably axiomatic that for real revision to take place, a piece of writing must have substance to it, a certain length. It is easier to revise a longer piece; there is more that can be deleted or rearranged.

Ellen Nold, who discusses the importance of the revision process states that "recent research indicates that one of the major differences between skilled adult writers and unskilled adult writers is the way they revise." [19] Citing Nancy Sommers' work on revising, she says that skilled adults revise globally first and then locally. [20] She makes reference to Beach's suggestion that the sophistication of a writer's revising strategies would be a good indication of the developmental level of the writer. [21]

Using the computer to word process, however, is not without its problems. There seem to be frustrations in learning any word processing program. It takes a period of time before the computer becomes for the writer an extension of his or her body in the same way as the pen usually is. It is possible, while one is learning a word processing program, to get snared in the web of its procedures and to lose important ideas and concentration, disturbing the rhythm and flow of the writing at hand.

In addition to mechanical interruptions, there can be logistical ones. If the computer is located in the user room of a school, there are the distractions inherent in the presence of others that may disturb the quiet concentration necessary for many writers. The writer may find he or she needs materials or sources that are not easily transported, such as dictionaries. The user room schedule may be inconvenient or the computers may be occupied when the writer wants to work. The computer may be "down" for repairs or for maintenance.

But most obstacles can be overcome. When the word processing program is mastered, many writers find that computers allow them to catch their idea flow faster and more efficiently than by pen. Once the idea has been captured, one of the great advantages the word processor represents to the writer over the pen is its ability to delete, to insert, and to move small or large chunks of information easily. Revision can be done swiftly. As one professional writer put it, "It takes the pain out of writing." The ease of revision encourages writers to go back over their work again and again, making words more effective, sentences more powerful, paragraphs more unified. One of the many questions that needs to be looked at more carefully is whether revision done on computers remains the same as might be expected from the writer's developmental level or if the process of writing on the computer facilitates the acquisition of more sophisticated techniques. This is important since improved revision strategies should result in improved writing.

The use of text editors to help writers, after they have word processed their writing, to eliminate errors of grammar and spelling, improve word choice and usage, even to point out organizational matters, is a way the computer is being used as an editing tool. One of the most extensive programs of this nature, developed at Bell Laboratories, is called the Writer's Workbench. These programs may be used by the writer to highlight potential problems. Given the highly complex nature of language, the computer will not always be right, however, and the writer makes the decision whether to change a feature or leave it alone. Conclusions to trials conducted by Bell Labs suggest their programs may result in improved writing. "Compared to first drafts, the last drafts of documents had fewer passive sentences, fewer abstract words, and fewer awkward or wordy phrases." [22] However, no control group was used. Without a control group it becomes impossible to know how much of the improvement on the final drafts is attributable to the programs and how much to the writer's own skill at editing. Most writers' final drafts will be a noticeable improvement over their first, even without a text editor. While these programs appear to be valuable aids to the writer, more research needs to be carried out which shows the exact nature of advantages and disadvantages in using them.

The composing process via computer may also affect a writer's style. Assessing style, "style in the sense of what is distinguished and distinguishing," as William Strunk and E.B. White put it, is difficult.

> Here we leave solid ground. Who can confidently say what ignites a certain combination of words, causing them to explode in the mind?...These are high mysteries. ...There is no satisfactory explanation of style. [23]

These difficulties of assessment suggest that comparing changes in a writer's style writing on and off the computer may not be made easily with any sense of objectivity. It seems likely such changes as well as other changes in the written product do occur although we may not achieve a good understanding of them for some time to come.

What is the future of the computer in the teaching of writing? In spite of the recent wave of enthusiasm which is greeting the computer in education, there are notes of caution. Alfred Bork states,

> It is not clear that the computer is going to improve education. The computer, like any new technology, has the potential for improving education or weakening education....the computer is a gift of fire. [24]

If we resolve the Great Writing Debate and the Great Computer Debate and agree that for teaching writing the computer is best used holistically as a tool rather than as a drill and practice instrument, the debates will still not be ended. The question of how the composing process is affected using the computer is a complex one, not readily answered. Work in the area of computers and writing needs to take into account the ongoing research on the writing process. Both teachers and researchers need to have a solid theoretical foundation on which to build their strategies and approaches for using the computer as a writing tool. Then, tempering our enthusiasm with a touch of caution, we should attempt to discover through research and personal experience the best educational implementations of this exciting new writing tool.

NOTES

[1]Seymour Papert, Mindstorms: Children, Computers and Powerful Ideas (New York: Basic Books, 1980), pp. 30-31.

[2]James Collins, "Speaking, Writing, and Teaching for Meaning," Exploring Speaking-Writing Relationships, ed. Barry M. Kroll and Roberta J. Vann (Urbana, IL: NCTE, 1981), p. 201.

[3]Anthony Petrosky, "Grammar Instruction," English Journal, 66 (December 1977), pp. 86-88.

[4]Sondra Perl, "The Composing Processes of Unskilled College Writers," Research in the Teaching of English, (December 1979), p. 109.

[5]Sondra Perl, "Understanding Composing," College Composition and Communication, (December 1980).

[6]Linda Flower and John Hayes, "Plans that Guide the Composing Process," in Writing: The Nature, Development and Teaching of Written Communication, Vol. 2, ed. Karl H. Frederiksen and Joseph F. Dominic (Hillsdale, NJ: Lawrence Erlbaum, 1981), p. 51.

[7]Flower and Hayes, "Plans," p. 51.

[8]Janet Emig, "Writing as a Mode of Learning," College Compositions and Communication, 28 (1977), pp. 122-128.

[9]Petrosky, "Grammar," p. 88.

[10]Emig, "Writing," p. 128.

[11]Walter J. Ong, "Transformations of the Word," in Interfaces of the Word (Ithaca, NY: Cornell University Press, 1977), p. 17.

[12]Ong, "Transformations," p. 17.

[13]Jack Goody and Ian Watt, "The Consequences of Literacy," in Literacy in Traditional Societies, ed. Jack Goody (Cambridge, England: Cambridge Univ. Press, 1968), p. 55.

[14]Hugh L. Burns and George H. Culp, "Stimulating Invention in English Composition Through Computer-Assisted Instruction, Educational Technology, 20, 8, p. 5. ERIC EJ 232 548.

[15]Burns and Culp, "Stimulating Invention," p. 9.

[16]Earl Woodruff, Carl Bereiter and Marlene Scardamalia, "On the Road to Computer Assisted Compositions," Journal of Educational Technology Systems, 10, 2 (1981-82), p. 141.

[17]Woodruff, Bereiter and Scardamalia, p. 142.

[18]Colette Daiute, "Word Processing," Electronic Learning, (March/April 1982), pp. 29-31.

[19]Ellen Nold, "Revising," in Writing, Vol. 2, ed. Carl H. Frederiksen and Joseph F. Dominic (Hillsdale, NJ: Lawrence Erlbaum, 1981), p. 67.

[20]Nold, "Revising," p. 67, citing N. Sommers, "Revision Strategies of Experienced Writers and Student Writers," MLA, December 1978.

[21]Nold, "Revising," p. 67, citing R. Beach, "Self-Evaluation Strategies of Extensive Revisers and Nonrevisers, College Composition and Communication, 27 (1976), 160-164.

[22]Nina H. MacDonald et al., "The Writer's Workbench," TEEE Transactions on Communications, Com. 30, No. 1 (1982), p. 109.

[23]William J. Strunk and E.B. White, The Elements of Style, Third Edition (New York: MacMillan, 1979), p. 66.

[24]Alfred Bork, "Reactions," in Computers in Composition Instruction, ed. Joseph Lawlor (Los Alamitos, CA: SWRL Educational Research and Development, 1982) p. 73.

REFERENCES

Bork, Alfred. "Reactions." In Computers in Composition Instruction. Ed. Joseph Lawlor. Los Alamitos, CA: SWRL Educational Research and Development, 1982, pp. 67-74.

Burns, Hugh L. and George H. Culp. "Stimulating Invention in English Composition Through Computer-Assisted Instruction." Educational Technology, 20, No. 8 (1980), pp. 5-10. ERIC EJ 232 548.

Collins, James L. "Speaking, Writing, and Teaching for Meaning." In Exploring Speaking-Writing Relationships: Connections and Contrasts. Ed. Barry M. Kroll and Roberta J. Vann. Urbana, IL: NCTE, 1981, pp. 198-214.

Daiute, Colette. "Word Processing: Can It Make Even Good Writers Better?" Electronic Learning (March/April 1982), pp. 29-31.

Emig, Janet. "Writing as a Mode of Learning." College Composition and Communication, 28 (1977), pp. 122-128.

Flower, Linda and John Hayes. "Plans that Guide the Composing Process." In Writing: The Nature, Development and Teaching of Written Communication, Vol. 2. Ed. Carl H. Fredericksen and Joseph F. Dominic. Hillsdale, NJ: Lawrence Erlbaum Assoc., 1981, pp. 39-58.

Goody, Jack and Ian Watt. "The Consequences of Literacy." In Literacy in Traditional Societies. Ed. Jack Goody. Cambridge, England: Cambridge Univ. Press, 1968, pp. 27-68.

MacDonald, Nina H., et al. "The Writer's Workbench: Computer Aids for Text Analysis." IEEE Transactions on Communications, Com. 30, No. 1 (1982), pp. 105-110.

Nold, Ellen W. "Revising." In Writing: The Nature, Development, and Teaching of Written Communication, Vol. 2. Ed. Carl H. Fredericksen and Joseph F. Dominic. Hillsdale, NJ: Lawrence Erlbaum Assoc., 1981, pp. 67-79.

Ong, Walter J. "Transformations of the Word." In Interfaces of the Word. Ithaca, NY: Cornell Univ. Press, 1977, pp. 17-36.

Papert, Seymour. Mindstorms: Children, Computers, and Powerful Ideas. New York: Basic Books, 1980.

Perl, Sondra. "The Composing Processes of Unskilled College Writers." Research in the Teaching of English, 13 (1979), pp. 317-336.

________. "Understanding Composing." College Composition and Communication, (December 1980), pp. 363-369.

Petrosky, Anthony R. "Grammar Instruction: What We Know." English Journal, 66 (1977), pp. 86-88.

Strunk, William, Jr. and E.B. White. The Elements of Style, Third Ed. New York: MacMillan, 1979.

Woodruff, Earl, Carl Bereiter, and Marlene Scardamalia. "On the Road to Computer Assisted Compositions." Journal of Educational Technology Systems, 10, 2 (1981-82), pp. 133-148.

SUGGESTIONS FOR FURTHER READING

Barth, Rodney J. "ERIC/RCS Report: An Annotated Bibliography of Readings for the Computer Novice and the English Teacher." English Journal, (January 1979), pp. 88-92.

Bell, Kathleen. "The Computer and the English Classroom." English Journal, 69 (1980), pp. 88-90.

Brandt, Ron. "On Reading, Writing, and Computers: A Conversation with John Martin Henry." Educational Leadership, 39, No. 1 (1981), pp. 60-64. ERIC EJ 253 758.

Bridwell, Lillian, Paul Reed Nancarrow and Donald Ross, Jr. "The Writing Process and the Writing Machines: Current Research on Word Processors Relevant to the Teaching of Composition." In New Directions in Composing Research. New York: Guilford Press. In press.

Burns, Hugh L. "A Writer's Tool: Computing as a Mode of Inventing." Paper, New York College English Association Conference, Saratoga Springs, NY, 3-4 Oct. 1980. ERIC ED 193 693.

Collins, Allan, Bertram C. Bruce and Andee Rubin. "Microcomputer-Based Writing Activities for the Upper Elementary Grades." In Proceedings of the Fourth International Learning Technology Congress and Exposition. Warrenton, VA: Society for Applied Learning Technology, 1982.

Cronnell, Bruce and Ann Humes. "Using Microcomputers for Composition Instruction." Paper, Conference on College Composition and Communication, Dallas, TX, March 1981. ERIC ED 203 872.

Daiute, Colette and Robert Taylor. "Computers and the Improvement of Writing." In The Association of Computing Machinery Proceedings. Los Angeles, November 1981.

Daiute, Colette. Writing and Computers. Reading, MA: Addison Wesley, in press.

Gingrich, Patricia S. "Writer's Workbench: Studies of Users." To appear in The 29th International Technical Communications Conference Proceedings. Boston, 5-8 May 1982.

Hennings, Dorothy Grant. "Input: Enter the Word-Processing Computer." Language Arts, LVIII (1981), pp. 18-22.

Hereford, Nancy-Jo. "Computers are Objects to Think With." Instructor, 91, 7 (1982), 86-89. ERIC EJ 258 782.

Horodowich, Peggy Maki. "Developing Stylistic Awareness on the Computer: A Tagmemic Approach." Paper, 21st Annual Meeting of the Midwest Modern Language Association, Indianapolis, 8-10 November, 1979. ERIC ED 198 530.

Jaycox, Kathleen M. Computer Applications in the Teaching of English. The Illinois Series on Educational Applications of Computers, No. 19e. Urbana, IL: Dept. of Secondary Ed., Univ. of Illinois, 1979.

Macdonald, Nina H. "The Writer's Workbench Programs and Their Use in Technical Writing." To appear in The 29th International Technical Communications Conference Proceedings. Boston, 5-8 May 1982.

Marcus, Stephen. "Compupoem: A Computer-Assisted Writing Activity." English Journal, 71, 2 (1982), pp. 96-99. ERIC EJ 257 703.

Rubin, Andee. "The Computer Confronts Language Arts: Cans and Should for Education." To appear in Classroom Computers and Cognitive Science. New York: Academic Press, 1983.

Schwartz, Helen J. "A Computer Program for Invention and Feedback." Paper, The 33rd Annual Meeting of the Conference on College Composition and Communic tion, 18-20 March 1982. ERIC ED 214 177.

________. "Monsters and Mentors: Computer Applications for Humanistic Education." College English, 44, 2 (1982), pp. 141-152. ERIC EJ 257 857.

________. "Teaching Stylistic Simplicity with a Computerized Readability Formula." Paper, The International Meeting of the American Business Communication Assoc., Washington, D.C., December 1980. ERIC ED 196 014.

Wisher, Robert A. "Improving Language Skills by Computer." Paper, The Annual Meeting of the Association for the Development of Computer Based Instructional Systems, Dallas, 1-4 March 1978. ERIC ED 165 710.

Woodruff, Earl, Carl Bereiter, and Marlene Scardamalia. "On the Road to Computer Assisted Compositions." Journal of Educational Technology Systems, 10, 2 (1981-82), pp. 133-148.

Wresch, William. "Prewriting, Writing, and Editing by Computer." Paper, The 33rd Annual Meeting of the Conference on College Composition and Communication, San Francisco, 18-20 March 1982. ERIC ED 213 045.

COMPUTERS IN EDUCATION: ACTIVITIES AT THE FEDERAL LEVEL

Dorothy K. Deringer

National Science Foundation*

*The views expressed herin are those of the author and do not necessarily reflect those of the National Science Foundation.

I. Introduction

A. The United States is changing from an industrial to an information society; computer technologies stimulate and facilitate this restructuring.

B. If we have structural job changes, should we have structural educational changes? If so, what kind?

II. What activites are occurring at the Federal level regarding mathematics, science, computers and education?

A. Commissions and Reports

1. National Academy of Sciences Convocation
2. National Commission on Excellance in Education
3. National Science Foundation Precollege Commission on Science, Mathematics, and Technology Education

B. Congressional Actions: In May, there were 20 bills in the Congress on mathematics and science education. The current status of these bills will be discussed.

C. Department of Education

D. National Science Foundation: NSF program guidelines were reviewed by the National Science Board in May. NSF science education programs and guidelines for proposal preparation in Fiscal Years 1983 and 1984 will be discussed.

III. Ten Projects in Mathematics and Science Using Computers

A. Logo--Seymour Papert, Massachusetts Institute of Technology

B. Computer Power--J. Michael Moshell, U. of Tennessee, Knoxville

C. Green Globs--Sharon Dugdale, U. of Illinois, Urbana

D. MathFun!--William Kraus, Wittenberg University

E. Algebra and Trigonometry--Gerald Issacs and Richard O'Farrell, Carroll College

F. Logic Tools--Ann Piestrup, The Learning Company

G. My Students Use Computers--Beverly Hunter, HumRRO

H. ComputerTownUSA!--Ramon Zamora, People's Computer Company

I. Math Network Curriculum Project--Diane Resek, San Francisco State University

J. The Puzzle of the Tacoma Narrows Bridge Collapse--Robert Fuller, U. of Nebraska, Lincoln

IV. The Future

KEEPING UP WITH COMPUTERS IN EDUCATION
or
COMPUTER PERIODICALS: PAST, PRESENT AND FUTURE

David H. Ahl
President, Creative Computing

Although many of the first digital computers were developed in the early 50's at various colleges and universities, it was not until the advent of minicomputers (1965) and microcomputers (1975), that the use of computers spread to all levels of education from the primary level through graduate schools. Today applications span a broad range and it is increasingly difficult to keep up with the latest developments.

One of the best ways of keeping up to date, perhaps the only way, is by means of computer periodicals. But where to start?

A Brief, Biased History

The first periodicals covering the use of computers in education were publications of two professional organizations, the Association for Computing Machinery (ACM) and Association of Educational Data Systems (AEDS). In 1967, the ACM Special Interest Group on Computers in Education started publishing a quarterly newsletter, the SIGCUE Bulletin. The major content was (and is) scholarly papers about computer usage in higher education.

Except for sponsoring an annual student programming contest, the major focus of AEDS is on administrative applications of computers, although in recent years the organization has broadened out somewhat to encompass instructional applications as well. The AEDS Monitor was first published in 1962, the AEDS Journal in 1967, and the AEDS Bulletin in 1974.

The first periodicals covering instructional applications at the secondary school level made the scene in the early 70's. Two were sponsored by the major manufacturers of minicomputers and small timesharing systems, Hewlett Packard and Digital Equipment Corp. The H-P Educational User Group Newsletter (first issue October 1970) was published monthly and then quarterly (more-or-less) until H-P's position in the educational market started eroding badly in 1977-78. DEC's newsletter, EDU, which I started in May 1971 was published when the mood was upon us, generally about four times a year. After a brief blackout when I left DEC in 1974, EDU resumed publication and has evolved today into one of the slicker manufacturer-backed educational newsletters.

A third early entry was People's Computer Company (first issue, October 1972), a tabloid newspaper put together almost entirely by Bob Albrecht. It carried some of the best programs and learning activities around, many of which are still in use today. Over the years, it evolved into a standard-size magazine called People's Computers (May 1977) and later Recreational Computing. It was finally absorbed by Compute! in Nov. 1981, an unfortunate death for one of the best early entries in the field.

Another early entry was Oregon Computing Teacher started by Dave Moursund in May 1974. Five years later, the name was changed to The Computing Teacher and it "went national."

Although EDU was geared to DEC's educational customers, by 1973 it had a circulation of 20,000 whereas DEC's educational customer base totaled less than 2000. While this difference was partially due to multiple copies going to the same schools, most of the additional subscribers were users of non-DEC computers and people who did not yet have systems. It occurred to me that perhaps the world needed an educational computer magazine that was not wedded to a single manufacturer; thus was born the idea for Creative Computing.

However, it was not until I left DEC in 1974 to take a position with AT&T, that I was actually able to start Creative. I started it initially as a hobby doing all the work on it (writing, editing, subscription processing, labeling, mail preparation, etc.) out of my basement. The first issue was published in November 1974. Like People's Computer Company, it was published on newsprint, although in standard magazine size with a heavy cover.

The next year was an incredibly important one: a little Albuquerque-based telemetry company, MITS, introduced a computer kit, the Altair 8800. Based on the Intel 8080 microprocessor, this was the first true microcomputer. The cover story in the January 1975 issue of Popular Electronics (now Computers & Electronics) propelled the sales of the Altair from MITS president Ed Roberts' projection of 400-500 to over 5000 in the very first year. Here, for the first time, was a computer that was within the reach of any school, hobbyist, or enthusiast.

In 1976, other kitmakers such as IMSAI, Processor Technology, Technical Design Labs, Parasitic Engineering, Southwest Technical Products, and Sphere sprang up like mushrooms. Most were started by enthusiastic engineers and lacked management depth and adequate financing. Some are still around today, but most died or were absorbed by other companies.

When there are products on the market with as little documentation and support as these early kits had, purchasers have a great need for additional information. Initially, computer kits were bought by hobbyists with some savvy in electronics, many of whom had experience in ham radio. Hence, most of the ham magazines started to run computing sections. One of these, Wayne Green's 73 Magazine, finally broke off its computer section as a separate magazine, Byte (first issue, September 1975).

Another source of information for hobbyists was computer clubs, some of which started publishing sizeable newsletters. The two largest clubs, the Amateur Computer Group of New Jersey and the Southern California Computer Society had major newsletters. As time went on, the SCCS, somewhat unbelievably, started forming chapters in other states (for those who wanted to live in Southern California but couldn't afford it?) and their newsletter (first issue, September 1975) three months later became a glossy magazine, SCCS Interface.

Nine months later, SCCS Interface was taken over by Bob Jones who renamed it Interface Age (first issue, September 1976). This apparently did not have the wholehearted blessing of the SCCS since they continued to publish SCCS Interface, renaming it Microcomputer SCCS Interface in August 1977. The second and final issue under that title was published a month later.

Another early entry, in January 1977, was Personal Computing. It was financed by Will Buchbinder, president of Benwill Publishing Co. in Boston, but the publisher, David Bunnell (formerly MITS' advertising manager), and editor, Nels Winkless, were located in Albuquerque. This arrangement was less than satisfactory, and eventually Benwill moved the magazine to Boston.

Unfortunately, all was not well at the 73/Byte empire in Peterborough, NH, and in late 1976 Wayne Green and his wife, Virginia, got a divorce. Wayne kept 73 Magazine (then much larger than the fledgling Byte) and Virginia took Byte. Well, Wayne really wanted a microcomputer magazine so he went the startup routine all over again, this time with Kilobyte (a little double meaning there). Virginia didn't think much of that so she commissioned a comic strip (one of the worst ever seen) under the title of Kilobyte. It served its purpose: published in Byte months before Kilobyte magazine was scheduled off press, it secured the copyright on the name Kilobyte. Hence, Wayne was forced to change the name of his new magazine to the somewhat meaningless Kilobaud (1000 bauds?). The first issue appeared in January 1977.

Two other microcomputer magazines were also started in 1977, Microtrek and ROM. Microtrek lasted two issues--it was slick and nicely done--but there just wasn't enough advertising to go around. ROM did somewhat better, lasting nine issues until it was merged into Creative Computing in late 1978. ROM was published by Eric Sandberg-Diment out of a barn in Hamden, CT which is not a commentary on its content. In fact, ROM failed for a novel reason--it was too far ahead of its time. It assumed that its readers knew nothing about computers and every article was a virtual tutorial. Published today, ROM would have a huge circulation and could compete with anything on the market, but in 1977 computer enthusiasts didn't want that beginner stuff.

People's Computer Company also started a magazine in May 1977, Calculators/Computers, which focused on instructional applications in the primary and middle school grades. Like People's Computers, it had excellent content but, alas, not enough advertising; the last issue was October 1979.

Another 1977 start was Computers and Education, a scholarly, refereed journal published by Pergamon Press. It is the only international journal, but its price ($40 for four issues) certainly holds down its circulation.

From 1978 to the present day, there has been an average of 11 new magazine startups per year with the pace accelerating in the last two years. Many, of course, have not lasted after the initial capital ran out, but the majority are still around today.

Where Are They Now?

You may be interested in what happened to some of the early entries cited above. The ACM and AEDS publications are still alive but have turned increasingly inward to the membership of their own organizations. I hate to call them dull, but a better word doesn't leap to mind.

After going slick on its two-year anniversary (November 1976), Creative Computing started to cover the microcomputer field as well as the educational area and carved out a niche as the leading magazine in applications and software. Today, its 300 plus pages per issue are filled with in-depth reviews, programs, tutorials, and "think pieces." It also has an active book publishing division. Since being acquired by Ziff-Davis Publishing Co. in December 1981, circulation has soared to over 300,000.

Popular Electronics, another Ziff-Davis magazine, changed its name to Computers & Electronics in October 1982 to be more consistent with its editorial content. From a five-year plateau, its newsstand circulation immediately rose by 20% and has been steadily increasing ever since.

Right from the start, Byte focused on computer hardware (technology, build-it-yourself, reviews) which remains its major focus today. It quickly became the bible of not only hobbyists, but OEMs and computer manufacturers as well. Hence, it carries a phenomenal amount of hardware advertising. Purchased by McGraw Hill in late 1979, it has continued to grow in circulation and bulk (500+ pages per issue).

Interface Age has suffered over the years from the lack of a consistent editorial direction. Initially a hobbyist magazine, it then shifted toward home uses, then to education, and, most recently to business applications. It frequently presents tables of product specifications, one of its more valuable features.

Personal Computing has also suffered from several shifts in editorial emphasis as it became a pawn in the magazine acquistion game. Now with its third owner, Hayden Publishing, it appears that Personal Computing, despite its name, is trying to appeal mainly to a novice business user with brief forays into home, educational, and other applications.

Right from the start, Kilobaud tried to be a layman's version of Byte with easier technical articles and more how-to pieces. Incidentally, Microcomputing was added to the name, Kilobaud, in January 1979. The Kilobaud (a horrible name) shrank, and finally disappeared altogether in January 1982. Unfortunately, Wayne Green is an extremely opinionated and abrasive publisher; this has detracted from the long-term success of Microcomputing even though the articles are often quite good.

Today, Where Does One Turn?

Today, there are over 70 magazines and journals of direct or indirect interest to people using computers in education. Even though a library or well-financed computer center may be able to subscribe to all of them, there just aren't enough hours in the day to read all of them. Hence, what I have done here is to list 30 categories of magazines. You should easily be able to narrow down to the three or four categories of interest to you and then subscribe to some or all of the magazines in those categories. Naturally, many magazines are listed in several categories, so this will help you narrow down also.

At the end, each magazine is listed with its address, publication frequency, and subscription rate. Before sending in your check, it is best to check the price since they tend to change fairly rapidly.

General Categories

Educational Applications
- ACM SIGCUE Bulletin
- AEDS Publications
- Classroom Computer News
- Computers and Education
- Computers in the Classroom
- The Computing Teacher
- Creative Computing
- Educational Computer
- Educational Technology
- [illegible]
- Instructional Innovator
- Media & Methods
- School Microware Reviews
- T.H.E. Journal

Recreational Applications
- Compute!
- Creative Computing
- Microcomputing
- Popular Computing
- Power Play
- Softline
- Softside

How-to Hardware Construction
- Byte
- Computers & Electronics
- Microcomputing
- Radio Electronics

Programming Techniques
- Creative Computing
- Softalk
- Softside

In-depth, Objective Reviews
- Byte
- Computer Update
- Creative Computing
- Creative Buyer's Guides
- Infoworld
- Infoworld Report Card
- Softalk

Index to Publications
- Microcomputer Index

Library Applications
- Access
- CMC News
- Computer Equipment Review
- Small Computers in Libraries

Home Applications
- Compute!
- Creative Computing
- Microcomputing
- Personal Computing
- Popular Computing

Business Applications
- Desktop Computing
- The Power of: ES
- Interface Age
- Personal Computing
- SATN

Communications Applications
- Today

Technical--Hardware
- Byte
- Computer
- Computers & Electronics
- IEEE Micro

Technical--Software
- Dr. Dobbs Journal
- Microsystems

Industry News
- Byte
- Infoworld
- The Jeffries Report

Of Interest to Beginners
- Digit
- MicroDiscovery
- Popular Computing

Of Interest to Young Readers
- CompuKids
- Creative Computing
- Digit

Computer Specific Categories

Apple Only
- Apple Orchard
- Call A.P.P.L.E.
- InCider
- Nibble
- Peelings II
- Softalk

Apple Column or Programs
- Compute!
- Creative Computing
- Micro
- Softside
- Softline

Atari Only
- Analog
- Antic

Atari Column or Programs
- Compute!
- Creative Computing
- Micro
- Softside

Color Computer Only
- The Color Computer
- Color Computer News
- Rainbow

Color Computer Coverage
- Creative Computing
- 68 Micro

Commodore Only
- Commodore
- Power Play

Commodore Column or Programs
- Compute!
- Creative Computing
- Micro

Heath/Zenith Only
- Buss
- REMark
- Sextant

Texas Instruments Only
- 99'er

IBM Only
- PC
- PC World
- Personal Computer Age
- Softalk for IBM PC

IBM Column or Programs
- Creative Computing

Timex/Sinclair Only
- Sync
- Syntax

Portable Computers
- Portable Computing
- Portable Companion
- Portable 100

TRS-80 Only
- 80 Micro
- 80-U.S. Journal

TRS-80 Column or Programs
- Creative Computing
- Softside

Names, Addresses, and Subscription Rates

Access: Microcomputers in Libraries P.O. Box 764 Oakridge, OR 97463 Quarterly, $11/year	Public access and stand-alone clerical applications. Many "think" pieces.
ACM 1133 Avenue of the Americas New York, NY 10036 Quarterly, $10/yr for members	ACM Special Interest Group on Computer Uses in Education publishes newsletter of spotty coverage of the field.
AEDS 1201 Sixteenth Street, N.W. Washington, DC 20036 Monitor, $22/6 issues Journal, $25/4 issues Bulletin, $7/10 issues	Assn. of Educational Data Systems publishes 3 titles: Monitor is magazine--mostly news, short articles; Journal has long, in-depth pieces; Bulletin is 4-pg newspaper.
A.N.A.L.O.G. 565 Main Street Cherry Valley, MA 01603 Bi-monthly, $12/year	Atari only. Programs, reviews and technical tips.
Antic 297 Missouri Street San Francisco, CA 94107 Bi-monthly, $15/year	Atari only. Excellent tutorials, technical tips. Many programs and reviews.
Apple Orchard 908 George Street Santa Clara, CA 95050 9 issues/year, $19.50/year	Apple only. Published by Int'l Apple Core, an organization of Apple user groups. News, software tips, etc.
Business Computer Systems 221 Columbus Ave. Boston, MA 02116 Monthly, controlled circulation	Articles, features charts of hardware and software, success stories, new products.
Buss 716 E Street, S.E. Washington, DC 20003 Approx 20 issues/year, $20/12 issues	Heath/Zenith only. Articles, news, hardware and software tips.
Byte 70 Main Street Peterborough, NH 03458 Monthly, $21/year	Leading technical magazine in the field. Detailed descriptions of hardware, construction articles, in-depth reviews.
Call A.P.P.L.E. 304 Main Street, Suite 300 Renton, WA 98055 7 issues/year, $20/year	Published by active Apple user group in WA. News, hardware tips, programs, etc.

Classroom Computer News 341 Mt. Auburn Street Watertown, MA 02172 8 issues/year, $19.95/year	Primary/secondary orientation. How to select software, classroom success stories, some reviews and programs.
CMC News 515 Oak Street North Cannon Falls, MN 55099 3 issues/year, $3/year	Stands for computers in the media center. Short (14 page) nuts-and-bolts newsletter. One long piece, many shorties.
The Color Computer Magazine P.O. Box 468 Hasbrouck Heights, NJ 07604 Monthly, $24/year	Program listings, assembly language material, how-to modifications, reviews.
Color Computer News P.O. Box 1192 Muskegon, MI 49443 Monthly, $21/year	Chock full of long and short programs, how-to tips, new product info.
Commodore 487 Devon Park Drive Wayne, PA 19087 Bi-monthly, $15/year	Published by Commodore. Much of interest to owners of these computers, but no outside viewpoints.
CompuKids P.O. Box 874 Sedalia, MO 65301 Monthly, $16/year	Calls itself "Computer Magazine for Beginners." Simple programs, stories, news.
Compute! P.O. Box 5406 Greensboro, NC 26403 Monthly, $20/year	Atari, Commodore, Apple only. Good tutorial approach, many programs, software tips.
Computer 10662 Los Vaqueros Circle Los Alamitos, CA 90720 Monthly, $32/year	Journal of the IEEE Computer Society. Computer systems design and application. Highly technical.
Computer Equipment Review P.O. Box 405 Saugatuck Sta., Westport, CT 06880 Semi-annual, $85/year	Formerly, Library Computer Equipment Review. Articles describe rather than evaluate. Good compendium source.
Computers and Education Pergamon Press, Maxwell House Fairview Park, Elmsford, NY 10523 Quarterly, $40/year for individuals	An international refereed journal. Theoretical emphasis. Papers deal mostly with college and secondary level.
Computers & Electronics One Park Avenue New York, NY 10016 Monthly, $15.97/year	Formerly Popular Electronics. Many do-it-yourself hardware projects. Continuing coverage of audio, consumer electronics

Computers in the Classroom
3 Carlaw Avenue
Toronto, ON M4M 2Z9
10 issues/year, $15/year

Canadian magazine. Mostly news, success stories, articles, and reviews.

Computer Update
Boston Computer Society
Three Center Plaza
Boston, MA 02108
Bi-monthly, $20/year

Publication of the Boston Computer Society. Good, early reviews of products, particularly ones from New England. Regional orientation.

The Computing Teacher
Dept. of Computer Science
University of Oregon
Eugene, OR 97403
9 issues/year, $14.50/year

Formerly, Oregon Computing Teacher. Many "think" pieces. Elementary/secondary emphasis. Much on computer literacy.

Creative Computing
39 E. Hanover Ave.
Morris Plains, NJ 07950
Monthly, $24.97/year

Many in-depth evaluations. Serious applications for home and school. Columns on Apple, Atari, TRS-80, IBM, Commodore.

Creative Computing Buyer's Guides
39 E. Hanover Ave.
Morris Plains, NJ 07950
Two per year, $3.95 each

Spring: Software Guide
Fall: Guide to Computers and peripherals. Half new material, half updated reprints from mag.

Desktop Computing
80 Pine Street
Peterborough, NH 03458
Monthly, $25/year

Business applications. Seems somewhat out of the mainstream in articles, but good comparison charts.

Digit
P.O. Box 29996
San Francisco, CA 94129
Bi-monthly, $11.95/year

New magazine with main aim toward young users.

Dr. Dobbs Journal
1263 El Camino Real
Menlo Park, CA 94025
Monthly, $25/year

Lots of long, long program listings. Operating systems, assembly language, and hard-core hacker stuff.

Educational Computer
P.O. Box 535
Cupertino, CA 95015
Bi-monthly, $15/year

Success stories, how-to classroom applications, reviews.

Educational Technology
140 Sylvan Avenue
Englewood Cliffs, NJ 07632
Monthly, $49/year

More a scholarly journal than a magazine. Mostly papers and experimental results.

Electronic Learning
902 Sylvan Avenue
Englewood Cliffs, NJ 07632
8 issues/year, $17/year

Published by Scholastic. Lots of teacher contributions and reviews. Slick presentation.

The Power of: ES 3543 N.E. Broadway Portland, OR 97232 Bi-monthly, $18/year	Business software only. How-to articles, programs, tutorials, news. (ES stands for Essential Software and seems to be title.)
IEEE Micro 10662 Los Vaqueros Circle Los Alamitos, CA 90720 Quarterly, $12/yr to IEEE members	Design concepts, engineering theory, algorithms, choice of hardware, circuit design, software. Very technical.
InCider 80 Pine Street Peterborough, NH 03458 Monthly, $19.97/year	Wayne Green's Apple only magazine. More of a beginner orientation than others.
Infoworld 530 Lytton Avenue Palo Alto, CA 94301 Weekly, $25/year	News, views, and reviews. More of an industry newspaper than an end-user publication.
Infoworld Report Card 530 Lytton Avenue Palo Alto, CA 94301 Occasional, $3.95/copy	Reprints of reviews of hardware and software. Good, objective material.
Instructional Innovator 1126 16th Street N.W. Washington, DC 20036 Monthly, $18/year	Covers audio visual, computers, and other related technology for schools.
Interface Age 16704 Marquardt Ave. Cerritos, CA 90701 Monthly, $21/year	Has moved toward primarily a business orientation. Good comparison charts of hardware and software.
The Jeffries Report P.O. Box 6838 Santa Barbara, CA 93111 Monthly, $30/year	Industry gosssip and news by Ron Jeffries. Called "a personal view of computing," it is the best 8-pg news sheet around.
Media & Methods 1511 Walnut Street Philadelpnia, PA 19102 9 issues/year, $24/year	Aimed at elementary/secondary schools; covers audio/video as well as computers.
Micro P.O. Box 6502 Chelmsford, MA 01824 Monthly, $18/year	Bills itself, "The 6502/6809 Journal." Technical info about computers using 6502 & 6809: Apple, Vic, Atari, etc.
Microcomputer Index 2464 El Camino Real #247 Santa Clara, CA 95051 Quarterly, $30/year	One-line descriptions of all articles, reviews, and programs in 39 magazines. Very comprehensive and complete.

Microcomputing 80 Pine Street Peterborough, NH 03458 Monthly, $25/year	Do it yourself hobbyist orientation. Circuits to build, programs to run, condensed reviews from other magazines
MicroDiscovery P.O. Box 7500 Bergenfield, NJ 07621 Monthly, $19.75/year	For beginners. Articles, tutorials, news. Very low level and non-technical.
Microsystems 39 East Hanover Avenue Morris Plains, NJ 07950 Monthly, $24.97/year	Technical software magazine. Mostly CP/M, Unix, & operating system software. Longer programs, some reviews.
Nibble P.O. Box 325 Lincoln, MA 01773 8 issues/year, $19.95/year	Apple only. Lots of programs and assembly language material. Technical orientation.
PC 39 East Hanover Avenue Morris Plains, NJ 07950 Monthly, $24.97/year	IBM personal computer & clones Huge magazine with articles, reviews of peripherals and software.
PC World 555 DeHaro Street San Francisco, CA 94107 Monthly, $18/year	IBM personal computer only. Articles, stories, reviews about IBM PC and look-alikes.
Peelings II P.O. Box 188 Las Cruces, NM 88001 9 issues/year, $21/year	Apple only. Programs, assembly language tutorials, reviews, short articles.
Personal Computer Age P.O. Box 70725 Pasadena, CA 91107 Monthly, $18/year	For IBM personal computer only. Amateur effort to cover IBM PC. Articles, hints, new products.
Personal Computing 50 Essex Street Rochelle Park, NJ 07662 Monthly, $18/year	Business orientation with some items for home and education. Many articles and success stories. Some reviews.
Popular Computing 70 Main Street Peterborough, NH 03458 Monthly, $18/year	Oriented to beginners. Nice graphic presentation, heavyweight writers (Asimov, etc.). Novice-level tutorials.
The Portable Companion 26500 Corporate Ave. Hayward, CA 94545 Bi-monthly, $12.50/year	Osborne computer only. Covers applications, success stories tips for using software.

Portable Computer 500 Howard Street San Francisco, CA 94105 Bi-monthly, $12.97/year	All portable computers. Covers applications, technology, trends, software, hardware, ideas.
Portable 100 Magazine Highland Mill Camden, ME 04843 Monthly, $24.95/year	Radio Shack Model 100 only. New from the publisher of Color Computer Magazine.
Power Play 487 Devon Park Drive Wayne, PA 19087 Quarterly, $10/year	Published by Commodore for Vic and Commodore 64 owners. Mostly home applications and games.
Radio-Electronics 200 Park Avenue South New York, NY 10003 Monthly, $13/year	How-to electronics hobbyist magazine. Covers entire field of electronics including personal computers.
Rainbow P.O. Box 209 Prospect, KY 40059 Monthly, $22/year	Radio Shack Color Computer only. Excellent tutorial material, programming techniques, many new product reviews.
REMark (Heath User's Group) Hilltop Road St. Joseph, MI 49085 Monthly, $18/year	Heath/Zenith computers only. Tutorials, reviews, Q & A, assembly language programming.
SATN P.O. Box 815 Qunicy, MA 02169 Bi-monthly, $30/year	Published by Software Arts for Visi-Calc users. Programs, hints, articles for users of Visi-Calc.
School Microware Reviews P.O. Box 246 Dresden, ME 04342 3 issues/year, $45/year	Collection of software reviews for elementary/secondary level. Best of its kind. Publishes directory also.
Sextant 716 E Street, S.E. Washington, DC 20003 Quarterly, $9.97/year	Heath/Zenith only. Programs, articles, news, tutorials.
Small Computers in Libraries Graduate Library School Univ. of Arizona, 1515 E. First St. Tucson, AZ 85721	Many current short articles, one or two long ones, some tutorial material.
Softline 11021 Magnolia Blvd. North Hollywood, CA 91601 Bi-monthly, $12/year	Computer games only. Interviews with game designers, high scores, reviews, playing hints.

Softalk
11021 Magnolia Blvd.
North Hollywood, CA 91601
Monthly, $24/year

Apple only. Cram full of articles, stories, reviews, programs, tutorials, hints, on Apple. Horrible layout.

Softalk for IBM Personal Computer
11021 Magnolia Blvd.
North Hollywood, CA 91601
Monthly, $24/year

IBM personal computer only. Programs, industry gossip, new product information.

Softside
6 South Street
Milford, NH 03055
Monthly, $30/year

Program listings for TRS-80, Apple, and Atari. Somewhat inner directed with little outside advertising.

Sync
39 East Hanover Avenue
Morris Plains, NJ 07950
Bi-monthly, $16/year

Timex/Sinclair only. The bible for this computer. Software tips, programs, reviews, hints, articles, tutorials, how-to.

Syntax
P.O. Box 457
Harvard, MA 01451
Monthly, $29/year

For Timex/Sinclair only. News, short programs, hints, letters.

T.H.E. Journal
P.O. Box 992
Acton, MA 01720
Bi-monthly, $15/year

Stands for "Technological Horizons in Education." Mostly success stories and articles.

Today
P.O. Box 639
Columbus, OH 43216
Bi-monthly, $18/12 issues

Published by CompuServe, an on-line data network. Describes new services and data bases.

68 Micro
5900 Cassandra Smith Road
Hixson, TX 37343
Monthly, $24.50/year

For 6800 and 68000 mpu only. Technical hints for owners of computers with this mpu. Programs, some reviews.

80 Micro
80 Pine Street
Peterborough, NH 03458
Monthly, $25/year

TRS-80 only. The bible for for Radio Shack owners. Programs, tutorials, hardware hints, stories, etc.

80-U.S. Journal
3838 South Warner Street
Tacoma, WA 98409
Monthly, $16/year

TRS-80 only. Technical information, programs, reviews for TRS-80 owners.

99'er
P.O. Box 5537
Eugene, OR 97405
Monthly, $25/year

Texas Instruments only. Hints, programs, news, reviews for TI owners.

THINKER TOYS

Craig Moore

Note:

Craig Moore makes use of seventeen microcomputers to teach computer elective classes to sixth, seventh, and eighth grade students. The students are a mixture of sex, race and ability. In most cases, the people he talks to who are interested in using computers in their classrooms, are looking for ideas on how to use the computer to teach traditional subjects, or how to use the computer to teach about computers. Mr. Moore would like to present his reasons for the use of computers to develop creative thinking.

* * *

When I first considered a topic for this paper, the conference title, "Computer: Extension of the Human Mind", seemed poorly chosen. I couldn't agree that a thousand dollar machine was, in any way, an extension of human mind. I was attributing the quality of mind, or intellect, to the computer.

Searching for a topic, I tried to identify the value of the computer ~~to~~ my classroom. For three years I've tried to make the most of the time I have with computer students. Why? What, of value, is being taught? While considering the merits of my instructional program, I experienced a flip-flop in my perception of the conference title to: "Computer...Extending the Human Mind". Perhaps that is what the title means. The computer gives us the opportunity to create a happening that reflects our decisions, values, aesthetics, humor, and genius. As a reflection of these mental processes, it is an extension of the human mind.

I then began to appreciate the choice of the conference title, for it had another interpretation. The computer is not only a projector, of the end results of our mental processes, it is also the tool that assists in extending our ideas.

For example, I ask students who are starting a programing project to outline their ideas. Most students, unless they are mimicking someone elses' work, can make an outline with one or two main ideas. When these students are asked to clarify what will happen when... or if.. or next, they admit that they hadn't thought of that. As students write programs and try them out on the computer, they eventually deal with the details of their idea. Their completed programs are often quite complicated. In this sense, the computer is extending their conceptualization.

I think that when the computer runs the student's program, it clarifies, for the student, the weaknesses in the idea and helps to suggest new directions of thinking. The process whereby a vague idea evolves into a working, observable, construction of events, i.e. a drawing of a house, a musical composition, or a video game, is the process of the computer extending the student's imagination and understanding of the problem. In my opinion, the ability to construct a model of an idea on a computer leads some students to make predictions, deductions, and logical constructions that would be beyond their ability by purely abstract reasoning.

In addition to being a sophisticated, electronic device for drafting ideas, the computer has the potential to revolutionize the sharing of information. While growing up, I felt sort of guilty when teachers told me that humans used only one-sixth of their brain. With the modern explosion of information, that same notion gives me hope that I have enough brain to cope. The computer can become the extension of our mind to all of the freely accessible information in the world.

Schools of the near future will be challenged by information networks that deliver information and instruction directly to the home. A student's progress will not be paced by age and grade. Home instruction will force the curricula and grouping of students, in school, to be much more flexible. Schools in the past, have been centers where information could be distributed to the masses. If computer networks usurp this role, what adjustments should be made by the schools? If schools maintain their present character, as meeting places for people to work together, I think it makes sense that they emphasize the development of interpersonal relationships. The school of the future should teach the skills of creativity and problem solving and give students practice in working as a member of a team.

One reason the computer appeals to middle school students is the magnetic nature of computer media. A magnetic creation can be redone without a trace of correction. Students can strive for a finished design that has not been limited by prior mistakes. As their skill permits, students will be able to generate modifications to their work as an aid to visualizing alternate solutions. The magnetic media facilitates design by successive approximation.

Another reason the computer appeals to middle school students, is linked to their age. Middle school students are at a difficult age. They are approaching adult type bodies, but they can't drive. They have basic computional skills, but no checkbook to try them on. For these students, operating a computer is an opportunity to be the boss, the person in the driver's seat. We should be careful to try and preserve this feeling when designing educational software.

I would like to see educational software capture and maintain student interest the way programming does. Some people might argue that programming and computer assisted instruction are innately different activities. Recognize though, that in both instances the student is operating a program that was written by someone else. We don't often think of students who are writing programs in BASIC (or some other high level language) as running a computer program. Yet, in fact, they are. We overlook this fact because in typical CAI applications, the computer tutors the student about a particular model, i.e using cross products to check the equivalence of fractions, or the computer checks the student's comprehension of a model with some drill and practices, i.e. identifying the states of the U.S.A..

More educational software has to be designed that will let students build their own models. Software that allows the student to make deductions, and construct conditions, execute those conditions according to some predefined rules, and gives the student feedback about the validity of those assumptions and constructions, will elevate the computer beyond its maligned role of the "electronic page turner" or a "hopped-up Systems 80".

MATHEMATICS EDUCATION AND COMPUTERS:
CAUSE FOR CONCERN OR CHANGE?

Paper Presented at

The Computer: Extension of the Human Mind II

July 22, 1983

Jerry Johnson
Department of Mathematics
University of the Pacific
Stockton, California

MATHEMATICS EDUCATION AND COMPUTERS:
CAUSE FOR CONCERN OR CHANGE?

Jerry Johnson

Let me begin with an apology. The stated title for this paper is by intent a disguise for its actual title "The Impact of Computers on Mathematics Education." My reasons for disguising the theme are three-fold. First, any discussion or conclusions concerning the impact of computers on mathematics education would seemingly require the extensive collection of educational data or the process of retrospection. Unfortunately, the appropriate computer technology has not been available long enough to justify either approach or especially to warrant any conclusions. Second, it is presumptuous for me to openly approach such a broad theme. Despite twelve years of involvement with the use of computers in the mathematics classroom, I remain a victim of the "forest-trees" syndrome. On occasion, I have caught glimpses of the "forest" but it is amongst the "trees" where I happen to live. Thus, I enjoy the opportunity to speculate but certainly to not claim to pontificate. Third, I am no longer sure that I understand the theme, especially the use of the word "impact." After consulting Webster's finest, I share an interesting linear chain of definitions and redefinitions:

IMPACT → FORCIBLY IMPINGING
↓
FORCIBLY INFRINGING ON
↓
FORCIBLY ENCROACHING ON
↓
(1) TO FORCIBLY ADVANCE BEYOND PROPER LIMITS
(2) TO TRESPASS UPON THE PROPERTY OR RIGHTS OF ANOTHER, ESPECIALLY STEALTHILY OR BY GRADUAL ADVANCES

To return to the stated topic of this paper, my intent is to examine the current excitement of mathematics educators regarding the possible uses of computer technology to enhance the learning of mathematics by students. I claim that this excitement is a cause for concern and will attempt to analyze if it is also cause for change. The perspectives used in my analysis will be that of history, educational research, curricular developments, available computer technologies, and a wet finger in the wind.

The Potential For Change

Change encompasses the processes of alteration, modification, deviation, and transformation. These processes are dynamic and continual. Within this context, the computer's potential for prompting changes in mathematics education can be interpreted from several perspectives:

1) Changes in the goals of mathematics education;
2) Changes in the mathematics curriculum;
3) Changes in the process of learning mathematics;
4) Changes in the process of teaching mathematics;
5) Changes in the process of doing mathematics; and
6) Changes in mathematics itself.

Many of these changes overlap and can have catalytic effects on each other.

These changes can occur in two ways--either unconsciously or consciously. Unconscious change is unintentional or as Webster would claim, while our mental faculties are asleep. The end result is that the changes are often haphazard and often temporary. One possible example of this type of change is the integration of calculators into the mathematics classroom. Many mathematics teachers have found themselves forced into the role of "protector" or "reactor" when they should be in the front assuming the role of "leader" or "director." As a result, no definitive results, changes, or direction have occurred. In some instances, mathematics teachers were the last to "know" of this integration and its potential for positive change.

In contrast, conscious change is intentional or with our mental faculties fully awake. The expected result is a series of changes which are controlled, purposeful, complete, and with extended effect. As to the computer's role within mathematics education, I suggest it is both possible and necessary to direct our excitement and concerns toward the purposeful and conscious coordination of potential changes.

The natural question is : But why change at all? Despite the current status of mathematics education, it is not good enough to claim that change-for-change's-sake is a viable justification. It is even wrong to point to declining mathematics scores, less rigorous requirements, and a declining morale amongst mathematics teachers. These are symptoms of a more serious problem for which the computer could only act as a temporary "bandaid" rather than a "miracle cure." Any decisions regarding changes should be rational and based on the computer's own merits, not from a position of pedagogical desperation or technological fascination.

I propose five "Conscious Reasons" for using the computer to promote change within mathematics education. These reasons are:

(1) Because computers are used by mathematicians to do mathematics;

(2) Because educational research supports the use of computers within the classroom;

(3) Because curricular trends indicate that mathematics educators may have no choice but to respond to the computer's presence;

(4) Because current computer technologies and software resources have great potential; and

(5) Because we will discover that computers can have great positive effect on the learning and teaching of mathematics.

The fact that I suggest these reasons as being "conscious" does not imply that they are obviously true or that they can be justified with metric tons of evidence. Thus, it is necessary to elaborate on each. Where the available evidence fails, I resume the right to temporarily hide behind a wet finger raised in the wind.

Conscious Reason One

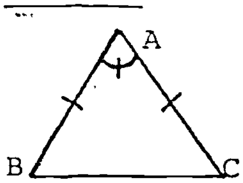
Thwaites (1970) contends that "the practice of mathematics is destined to be entirely changed by the computer. I do not exaggerate my own view of the future of mathematical methodology when I suggest that our present pencil-and-paper method will, in the historical perspective, be likened to the industrial methods of the stone age." Several examples can be offered to support Thwaites' claims that the "doing" of mathematics is changing.

As early as the 1700's, Leibniz expressed an interest in the creation and implementation of a general decision procedure for the proof of theorems. A similar interest was shown in the early 1900's by both Peano and Hilbert. In 1930, Herbrand developed a mechanical method but it expectedly was extremely tedious and time-consuming. Gilmore eventually adapted Herbrand's techniques in 1960 to implementation on a computer. However, Gilmore had little success because the computer techniques could only prove a formula (theorem) if its negation was inconsistent, which is often impossible to fully test. A major breakthrough in 1965 was Robinson's conceptualization of a "Resolution Principle" which introduced the use of a single inference rule (Loveland, 1978). Since 1965, the adapted computer techniques have been used to achieve some impressive results which unfortunately are beyond the realm of most mathematicians. That is, the efforts in this area have focused more on the improvement of the theorem-proving techniques via a computer rather than their application to the actual proof of new theorems within the domain of mathematics.

A related and more accessible example is the computer program LOGIC THEORIST by Newell and Simon (1963). Their intent was the development of a computer program which would be capable of proving theorems in the sentential calculus. It was tested using the initial assumptions and inferential rules stated in Principia Mathematica by Whitehead and Russell (1910). Their computer program essentially proved 38 of the first 52 theorems. Overall, the program proved 300 theorems in less than 10 minutes with some of the computer's proofs being regarded as more elegant than those provided by Whitehead and Russell. The most famous instance of an elegant proof is the computer program's proof for the following common geometric theorem:

<u>THEOREM</u>: The two base angles of an isosceles triangle are congruent.

<u>DIAGRAM</u>:



<u>PROOF</u>:

1. $AB = AC$
2. $AC = AB$
3. $\angle BAC = \angle CAB$
4. $\triangle ABC = \triangle ACB$

$\therefore$ 5. $\angle ABC = \angle ACB$

At first, many mathematicians were amazed at the simplicity of this "new" proof. Their amazement was shortlived, however, when someone pointed out that Pappus had used this same proof around 300 A.D.

A second example is the computer's application to the four-color problem, a conjecture which eluded mathematicians for over 125 years. Many of these mathematicians felt that the computer would also be of no use in this area. Lehmer (1968) claimed that "not all searches for counterexamples have triumphal endings. In fact, in the case of some famous unsolved problems, it would seem foolhardy to invest good machine time in such a search. Examples are the four-color problem...." Nonetheless, in 1976, Appel and Hagen (1980) used 1200 hours of computer time at the University of Illinois to finally solve the four-color problem: every planer map <u>can</u> be colored with four colors in a manner whereby each country sharing a common border receives a different color. The solution involved the extensive use of sets of reducible configurations and its correctness cannot be checked without the assistance of a computer. The nature of this non-traditional proof "by exhaustion" has raised concerns within the mathematics community and has led to many discussions as to what exactly constitutes a proof in mathematics (Tymoczko, 1980, 1981).

My final example involves variations of Fermat's Problem:

> Does the equation $x^3+y^3=z^3$ have a positive integer solution?

Euler extended the problem to these two equations:

$x^4+y^4+z^4=w^4$ and $x^5+y^5+z^5+w^5=v^5$

and again conjectured that no positive integer solutions would ever be found. However, in 1968, a computer was used to find a solution for his final equation:

$$27^5+84^5+110^5+133^5=144^5$$

In fact, the computer techniques used to find the solution can be understood and implemented by pre-college students.

These examples clearly demonstrate that the computer can be used to do mathematics though the approaches used are possibly different. Computer-generated data can be used in the mathematics classroom as well to suggest conjectures, destroy conjectures, and even prove some conjectures by exhausting all possible cases. The process is extremely active, as demonstrated by Atkin (1968) who attempted to extend congruence properties of modular forms using a computer: "It might in theory have been possible to arrive at some of these results by a priori reasoning; certainly the cases where the proofs are complete require in the event no more computation than a few hours' hand work. But I would not myself have been able to conjecture the possibility of such results without the evidence provided by the computer."

Conscious Reason Two

Considerable research has been completed relative to the various roles a computer can assume within a classroom, especially within mathematics education. The research has focused on these areas:

(1) CAI as a substitute for traditional instruction;
(2) CAI to augment traditional instruction;
(3) Effects of modes of CAI (Drill & Practice, etc.);
(4) Compression of instructional time;
(5) Retention of content;
(6) Interactional effects of ability levels; and
(7) Effect on and of attitudes.

Numerous surveys and meta-analysis studies have been completed regarding these various areas (Dence, 1980; DeVault, 1981; Edwards etal, 1975; Thomas, 1979; Burns & Bozeman, 1981; Anastasio & Morgan, 1972; Overton, 1981). The general conclusion of these surveys is that because of conflicting data, inconclusive results, uneven experimental designs, changing technologies, and a lack of comparative efforts, the verdict remains out on the effectiveness of a computer regardless of its role. Adding some necessary humor to a frustrating situation, Hubbard (1980) suggests that the studies have proven that the use of a computer is better than nothing; that is, results from a computer-using group while on task are always better than the results from non-task computer-using groups.

Nonetheless, some positive generalizations are both possible and relevant to the theme of this paper. First, Overton (1981) and Vinsoholer and Bass (1972) have concluded that one mode where the computer has proven its effectiveness is in the area of drill and practice. And second, using meta-analysis techniques for comparing the available studies collectively, Burns and Bozeman (1981) claimed "a significant enhancement in instructional environments supplemented by CAI, at least in one curricular area--mathematics."

The lack of convincing research results which could support the classroom use of computers can be extremely deceiving. Using my own classroom experiences and those described by others, I am convinced that the computer can have a powerful effect on the learning of mathematics. Perhaps this effect has not been measured by the past research or experimental methodologies. It is even possible that the computer was not being used in a manner where these powerful effects could be realized.

An example of a research project which is positive and exemplary is a study by Brooks (1970). He investigated the effects of using graphics in the teaching of topics such as equation root-finding, numerical approximation techniques, and the solution of ordinary differential equations. His pedagogical model involved the use of teacher-prepared examples, classroom-generated examples, and student laboratory experiences. The majority of his conclusions are appealing:

(1) The computer "stimulates more and better class discussions and questions;"

(2) The "material learned seems to be retained better";

(3) The students "attack examination questions by drawing a sketch first. Their solutions are more direct, with less floundering";

(4) "Instructor preparation time averages 4 hours per class hour";

(5) "Up to 25% more class time is required for each topic"; and

(6) Each class plan must include a fall-back plan to use in case of system failure.

The last three conclusions or observations can be countered by the availability of new and better computer technologies. Nonetheless, they are even perhaps small in cost relative to the positive implications of the study's other conclusions.

Conscious Reason Three

As early as 1954 W. Manheimer predicted that "in the long run the digital computer will cooperate with other modern ideas to revise and improve our mathematics syllabi." During the past thirty years, little progress has been made in the development of mathematics curricula which integrate the computer. It is neither enough nor appropriate to add two-page supplements to chapters in existing textbooks. No clear examples exist of curriculums or textbooks which fully capitalize on the computer's potential. The National Council of Teachers of Mathematics has not given the proper leadership with its issue of vague position statements, such as those found in the NCTM's Agenda For Action.

The greatest advances in the area of curriculum development have been prompted by an on-going debate within college mathematics departments. Focusing on whether a course in discrete mathematics should replace or merge with calculus as a college student's initial experience in college-level mathematics, the debate was initiated by Anthony Ralston (1980), a past president of the ACM. His arguments can be traced back, however, to a statement by Givens (1960):

> There is a simple and yet basic fact about a computer which will, in the decades and centuries to come, affect not so much what is known in mathematics as what is thought important in it. This is its finiteness....The deep, pervasive and enduring shift in the mathematical sciences is in the criteria for what is interesting and important in mathematics, namely, from the infinite to the finite and from the theorem to the algorithm.

In June 1982, a special Sloan Conference was held at Williams College for the purposes of developing sample curricula which would reflect this new look in mathematics.

If the college mathematics curriculum changes, the pre-college mathematics curriculum can also be expected to change. Presently, the sequence and content of secondary mathematics are justified as meeting the necessary prerequisites for the study of calculus. Thus, if calculus is supplanted from its current status, the secondary school mathematics curriculum can be expected to change radically. In November 1982, a NSF Conference was held at the University of Maryland for the purpose of drafting alternative secondary school mathematics curricula which reflect both the changing college-level mathematics curriculum and the computer's presence.

These developments and Conferences demonstrate that the potential for changes in the mathematics curriculum are being discussed. Unfortunately, too many unjustified assumptions are affecting the direction of the discussions. We must dare to ask the key question: What ought to be taught in mathematics? Answers should be posed and discussed separate from any influence which either existing mathematics curricula or the computer's presence can have on the decision-making process. The goal is a positive synergism with both entities combining their respective powers and knowledge bases to accomplish that which neither is able to do alone. The end result would truly be a new mathematics curriculum. Old topics would be presented in new ways or with new emphases. Tedious or routine manipulations would not be a requirement and thus no longer usurp a great amount of instructional time. Many old topics (some favorites perhaps) would be replaced by new topics on mathematics of greater importance and relevance.

Conscious Reason Four

Current computer technologies have many capabilities which can enhance the teaching of mathematics more than any other content area. One example is the computer's increasing potential for graphics options such as color, shading, rotation, animation, and hard-copy screen dumps. Another example is the computer's increased capability for number-crunching due to improvements in speed, ease of access, and precision. A third example, and possibly the most important, is the decrease in hardware costs which makes the concept of a powerful personal computer viable as a tool which can assist both the study and doing of mathematics. Adjusting for inflation, the costs of several microcomputers are currently below those of elaborate slide rules less than fifteen years ago.

New developments in computer software offer great promise for mathematics education as well. The best example is muMATH, a computer-symbolic mathematics package which is interactive and can execute almost any algebraic or analytic operation. muMATH performs exact rational arithmetic on variable expressions with almost 600-digit precision. It includes all of the mathematics neccessary to handle problems through the first-year calculus level.

To test its algebraic abilities, I used muMATH while taking a standardized examination designed for the end of an algebra course. To illustrate some of its potential, the examination problem will be stated, followed by its restatement in muMATH syntax, and finally followed by the obtained results as designated by the symbol "@":

(1) When 9 is added to 5 times a certain number, the square root of the result is 8. Find the number.

```
?SOLVE((9+5*X)Λ(1/2)==8,X);
@: {X==11}
```

(2) If x=m-2n and y=2m-n, find the value of 2x+y-(x-y) in terms of m and n.

```
?X:M-2*N;
?Y:2*M-N;
?2*X+Y-(X-Y);
@: 5*M-4*N
```

(3) Factor over the integers: a(m-1)-(m-1).

```
?FCTR(A*(M-1)-(M-1));
@: 1-(A+M)+A*M
```

(4) Perform the indicated operations: $(2a-3b)(a^2-5ab+6b^2)$

```
?EXPD((2*A-3*B)*(AΛ2-5*A*B+6*BΛ2));
@: 27*A*BΛ2-13*AΛ2*B+2*AΛ3-18*BΛ3
```

(5) For what values of x is this true: mx-x=5.

```
?SOLVE(M*X-X==5,X);
@: {X==5/(-1+M)}
```

(6) Solve for x: $\frac{x-2}{x+4} + \frac{x-1}{x-2} = \frac{15}{7}$.

```
?SOLVE((X-2)/(X+4)+(X-1)/(X-2)==15/7,X);
@: {X==3
    X==-40}
```

(7) For what values of x is this true: 3x+2=2(2x+1)-x.

```
?SOLVE(3*X+2==2*(2*X+1)-X,X);
@: {X==ARB(1)}
```

NOTE: ARB indicates a degenerate situation.

These problems illustrate that the solutions provided by muMATH are not always in the form that we would expect. Some of this discrepancy is due to the values given to the internal flags which control the transformations of the symbolic expressions. The discrepancy is also due to the program's internal use of different algorithms for certain processes such as factoring.

muMATH is but one of many computer-symbolic mathematics packages (e.g. MACSYMA, REDUCE, SMP) yet it is unique in being the first to be implemented successfully on a microcomputer (Wil[illegible], 1982; Engleman, 1971; Moses, [illegible] Pavelle etal, 1981). Other related types of mathe-[illegible]al software are currently being produced. For [illegible]ple TK SOLVER will complement muMATH in that it can handle decimal expressions, eliminates the need for the manipulation of algebraic formulae, and uses iterative procedures for solving equations.

In his interesting overview of the "Mechanization of Mathematics," Engleman (1971) envisioned "a mechanical assistant possessing an encyclopedic knowledge of mathematical formulae and results, capable of rapid and flawless computation, familiar with a broad spectrum of computational algorithms." In computer programs such as muMATH and TK SOLVER, that assistant is possibly already here.

Conscious Reason Five

The computer can improve the teaching and learning of mathematics. Its dynamic capabilities complement the broad goals of ma[illegible]hematics education which are too often not realized [illegible]use students become buried in the [illegible]udy of mathematical technique and the mastery of skills. Fortunately, the computer allows the student to transcend the rote and to approach mathematics on a conceptual level. The teaching approach required is radically different from today's norm with neither the current mathematics curriculum nor most mathematics teachers ready for its new focus. Nievergelt (1975) has aptly summarized the new methodology:

> The impact computers may have on mathematics education comes from the fact that they open new areas of study and allow new approaches to old problems and thus may greatly influence a student's image of the nature of mathematics....
> When a mathematical problem is approached with a view towards using a computer in its solution, the situation is very similar to that of an experimental scienti[illegible] [illegible] a laboratory experiment:... Looking for [illegible]ata...formulating hypotheses...and

checking both data and hypotheses. In this latter disguise, mathematics is much more accessible to many students who may be turned off when the subject is presented to them in more abstract ways.

The term "missing curriculum" was proposed by Magin (1976) to encompass those skills and concepts which are extremely important but have been avoided or reduced in potency because they have not been possible within the current instructional environment. A prime example of this missing curriculum is the solution of equations and inequalities. The current algebra curriculum devotes an inordinate amount of time and attention to the development of solution techniques (e.g. factoring, synthetic division, special transformation patterns) which are applicable to only a small set of cases (e.g. quadratics and a few special polynomials). To make matters worse, these techniques are quite mechanical and provide no real insight into the solution process as a generalizable process. In contrast, the computer and related software can provide access to a wide range of numerical solution procedures which are extremely powerful and are applicable to a wide range of cases. Furthermore, the techniques (e.g. finding roots by the Newton-Raphson method) not only provide insight into the solution process but are also quicker, more intuitive, and can be represented visually and simultaneously using computer graphics.

A comparison of techniques for the solution of systems of linear equations provides a second interesting example. The student is no longer forced to settle for the application of limited methods such as elimination or substitution to systems of three or fewer variables. Due to the computer's capabilities for number-crunching, powerful methods are now available which have previously been avoided because of their dependence on extensive computations. Examples are Gaussian elimination, the Gauss-Siedel method, Monte Carlo Techniques, and the use of Markov chains (Dorn & Greenberg, 1967; Seber; 1981; Nelson, 1977; Yakowitz, 1977). The numeric output and related graphics can now be analyzed step-by-step. Such a thorough analysis can enhance the student's understanding of the procedures while also promoting a feeling for the convergent sequential processes, algorithms, and geometry inherent within the solution of a system. Ill-conditioned systems can be used to determine the effects of round-off error and special assumptions associated with each procedure.

Many other subject areas within mathematics could be discussed. Examples include the construction and study of algorithms in place of proof and the geometrical intuitions developed by languages such as LOGO (Papert, 1980).

The Process Of Change

This paper began by raising an important question: Should the computer be used to prompt change within mathematics education? As partial responses to this question, several "Conscious Reasons for Change" have been discussed briefly. This still falls short , however. If an authority somewhere would (or even could) make the decision that changes in mathematics education must occur and involve the computer, there is no guarantee and little chance that the changes will occur let alone succeed as intended. Unfortunately, the dynamics of change are extremely complex and relatively misunderstood.

The Rand Corporation has studied federally-funded projects in education during the years 1973 through 1978. The study's intent was to identify the primary factors behind successful change and its lasting effects.[9] Though most of the study's conclusions are discouraging, four requirements have been suggested for any change process in education:

(1) Each local school site must serve as a center of change;

(2) There is a need for local grass-roots ownership of the change activities;

(3) The change agents should stress projects and activities that address real school-site problems; and

(4) Preference should be given to practical materials or materials generated by teachers and in-service training activities.

The Rand Study's requirements overlap in that classroom teachers become the key factor behind whether a change will be successful. The secondary factors are elements such as money, administrational support, and the validity of the idea itself.

The conclusions of the Rand Study can easily be interpreted within the context of promoting change within the mathematics classroom by using the computer as a change tool. However, the classroom teacher once again becomes the key factor. No amount of supportive research, clever software, or computer technology can succeed unless three conditions are met. First, the mathematics teacher must accept the use of a computer as being relevant to their own classrooms. Second, the mathematics teacher must identify the various uses of the computer within a classroom as being under his/her own direction. And third, the mathematics teacher must be involved from the start in the development of classroom activities using the computer to teach mathematics.

As [illegible] change process is lengthy and requires [illegible] of careful planning. [illegible]ort- projects [illegible] teacher-training sessions do little more than waste money, time, and motivation. Support, both financial and moral, must be found to initiate a large-scale process of change which is built, directed, and desired from within the ranks of the classroom teacher of mathematics. The nature of the desired changes and the impending computer technologies are such that there are no experts--rather we are all learners.

Thus, this is my course for change. It is also my cause for concern. As time pressures will accelerate the process, some form of change will occur and possibly not with the expected results. It is our responsibility as mathematics educators to make the best of what could either be our possible or an impossible situation.

> "There is a certain relief in change, even though it be from bad to worse; as I have found in travelling in a stage-coach, that it is often a comfort to shift one's position and be bruised in a new place."
>
> Washington Irving
> 1783-1859

77

Conscious Reason One:

Appel, K. & Haken, W. "The Four-Color Problem," in L. Steen (ed.) *Mathematics Today*. New York: Vintage Books, 1980. pp. 153-180.

Churchhouse, R. & Herz, J. *Computers in Mathematical Research*. North-Holland, 1968.
   (a) D. Lehmer. "Machines and Pure Mathematics." pp. 1-7.
   (b) C. Froberg. "On Some Number-Theoretical Problems Treated With Computers." pp. 84-88.

Lehmer, D. "Automation and Pure Mathematics," in W. Freiberger & W. Prager (eds.) *Applications of Digital Computers*. Ginn and Co., 1963. pp. 219-231.

Loveland, D. *Automated Theorem Proving: A Logical Basis*. North Holland, 1978. (Especially Chapter One).

Mendelson, N. "Some Examples of Man-Machine Interaction In The Solution of Mathematical Problems," in J. Leech (ed.) *Computational Problems in Abstract Algebra*. Pergamon Press, 1970. pp. 217-222.

Nevins, A. "Plane Geometry Theorem Proving Using Forward Chaining." *Artificial Intelligence*, 6(1975), pp. 1-23.

Newell, A. & Simon, H. "The Logic Theory Machine," in E. Feigenbaum & J. Feldman (eds.) *Computers and Thought*. McGraw-Hill, 1963.

Slagle, J. *Artificial Intelligence: The Heuristic Programming Approach*. McGraw-Hill, 1971. (Especially Chapters 4-6).

Stockmeyer, L. & Chandra, A. "Intrinsically Difficult Problems *Scientific American*, (May 1979), pp. 140-159.

Thwaites, B. "The Role of the Computer in School Mathematics." *Educational Studies In Mathematics*, 2(1969), pp. 346-359.

________. "Mathematics in 1984--The Impact of Computers," in F. Alt & M. Rubinoff (eds.) *Advances in Computers*, Vol. 10. Academic Press, 1970.

Tymoczko, T. "Computer Use to Computer Proof: A Rational Reconstruction." *Two-Year College Mathematics Journal*, (March 1981), pp. 120-125.

________. "Computers, Proofs, and Mathematicians: A Philosophical Investigation of the Four-Color Proof." *Mathematics Magazine*, 53(May 1980), pp. 131-138.

Whitehead, A. & Russell, B. *Principia Mathematica*, Vols. 1-3. Cambridge University Press, 1910-1913.

Conscious Reason Two:

Anastasio, J. & Morgan, J. Factors Inhibiting The Use Of Computers In Instruction. Princeton: EDUCOM, 1972.

Brook..., . "Computer-Man Communications: Using Computer Graphics in the Instructional Process," in F. Al... & M. Rubinoff (eds.) Advances In Computers, Vol. ... pp. 129-143.

Burns, P. & Bo..., W. "Computer-Assisted Instruction and Mathematics Ach... Is There a Relationship?" Educational Technology, (October 1981), pp. 32-39.

Dence, M. "Toward Defining The Role Of CAI: A Review." Educational Technology, (November 1980), pp. 50-54.

DeVault, M. "Computers," in E. Fennema (eds.) Mathematics Education Research: Implications For The 80's. NCTM/ASCD, 1981.

Edwards, J., Norton, S., Taylor, S., Weiss, M. & VanDusseldo R. "How Effective is CAI? A Review of The Research." Educational Leadership, 33(1975), pp. 147-153.

Hubbard, G. "Education and Training and the New Technologies in A. Howe (ed.) International Yearbook of Educational and Instructional Technology, 1980/81. Nichols Publishing Company, 1980. pp. 44-52.

Overton, V. "Research In Instructional Computing and Mathematics Education." Viewpoints In Teaching And Learning, (Spring 1981), pp. 23-36.

Thomas, D. "The Effectiveness of Computer-Assisted Instruction in Secondary Schools." AEDS Journal, 12(1979), pp. 103-116.

Vinsonhaler, J. & Bass, R. "A Summary of Ten Major Studies on CAI Drill and Practice." Educational Technology, 12 (1972), pp. 29-32.

Conscious Reason Three:

Atkinson, M. "Mathematics in the Service of Computer Programming." Mathematical Spectrum, 10(1977/78), pp. 6-11.

Birkhoff, G. "Mathematics and Computer Science." American Scientist, (1975), pp. 83-91.

Givens, W. "Implications of the Digital Computer for Education in the Mathematical Sciences." Communications of The ACM, (September 1966), pp. 664-666.

Knuth, D. "Computer Science and Its Relation to Mathematics. American Mathematics Monthly, (April 1974), pp. 323-342.

Manheimer, W. "The Digital Computer--A Challenge to Mathematics Teachers." School Science and Mathematics, (December 1954), pp. 701-706.

Ralston, A. "Computer Science, Mathematics, and the Undergraduate Curricula in Both." American Mathematical Monthly, (August/September 1981), pp. 472-485.

Conscious Reason Four:

Engleman, C. "The Mechanization of Mathematics," in Computers in Undergraduate Science Education Conference Proceedings. Chicago, August 17-21, 1970. pp. 103-116.

Wilf, H. "The Disk With The College Education." American Mathematical Monthly, (January 1982), pp. 4-8.

Moses, J. "Algebraic Simplification: A Guide For The Perplexed." Communications of the ACM, (August 1971), pp. 527-537.

Pavelle, R., Rothstein, M., & Fitch, J. "Computer Algebra." Scientific American, (December 1981), pp. 136-152.

Conscious Reason Five:

Artiaga, L. & Davis, L. Algorithms and Their Computer Solutions. Charles E. Merrill, 1972. (Especially the Chapters on Numerical Algorithms (#5), Polynomials (#6), the Solution of Equations and Plotting (#8), Sorting (#9), and Applications (#10)).

Dorn, W. & Greenberg, H. "Solving Equations By Iteration." in their Mathematics and Computing With FORTRAN Programming. John Wiley & Sons, 1967. pp. 270-323.

Forsythe, G. "Pitfalls in Computation: Why a Math Book Isn't Enough." American Mathematical Monthly, (November 1970), pp. 931-956.

Hamming, R. Calculus and The Computer Revolution. Houghton Mifflin, 1968.

Henney, A. & Henney, D. "Computer-Oriented Solutions." Crux Mathematicorum, 4(October 1978), pp. 212-216.

Kovach, L. Computer-Oriented Mathematics. Holden-Day, 1964.

Nelson, E. "A Machine-Oriented Technique for The Complete Solution of Linear Systems." Two-Year College Mathematics Journal, (June 1977), pp. 161-164.

Nievergelt, J. "Computers and Mathematics Education." Computers and Mathematics With Applications, 1(1975), pp. 121-132.

80

Magin, D. "Some Effects of Educational Technology On Curriculum Reform: Examples From the University Of New South Wales." TERC R&D Paper #43, (November 1976), ED 172716.

Stibitz, G. & Larrivee, J. Mathematics and Computers. McGraw-Hill, 1957.

Seber, R. "Systems of Linear Equations With Mini-Calculators or Computers." School Science and Mathematics, (October 1981), pp. 512-516.

Yakowitz, S. "Markov Chains and Linear Equations" in his Computational Probability and Simulations. Addison-Wesley, 1977. pp. 95-104.

The Process of Change:

Berman, P. & McLaughlin, M. Federal Programs Supporting Educational Change, Vols. 1-8. Rand Corporation, 1973-1978.

Frand, J. "A Strategy for Implementing Change of Mathematics Curricula." Elementary School Journal, (1978), pp. 118-123.

James, R. "Understanding Why Curriculum Innovations Fail or Succeed." School Science and Mathematics, (October 1981), pp. 487-495.

Milner, S. "Training Teachers About Computers: A Necessity for Education." Phi Deltan Kappan, (April 1980), pp. 544-546.

Price, J. & Gawronski, J. Changing School Mathematics: A Responsive Process. NCTM, 1981.

Trafton, P. "Assessing the Mathematics Curriculum Today," in M. Lindquist (ed.) Selected Issues in Mathematics Education. NCTM, 1980. pp. 9-28.

By Merrianne Metzger

The crisis in the field of special education is quickly coming to the point that school districts will need to seek new solutions in terms of service delivery or face a backlash of public disfavor over claims that even though funds were increased, clinician were unable to affect a change in the academic and/or social development of handicapped children. In other words our special education population has not decreased. This crisis is being brought to a head by the fact that programs mandated by P.L. 94-142 are tied legally to constitutional law and cannot be cut. In order to pass local school budgets, districts have reduced the funding for most other programs. However, since they are unable to cut special education programs thay have elected instead not to hire the additional special education staff members needed to meet the increased caseload created by P.L. 94-142 guidelines. In some cases school boards have been forced to cut positions in order to protect federally mandated programs and still present to the public a resonable budget for special education services. How did the special education crisis evolve? The following is a summary of the problem and a proposed solution involving special education applications of computer technology.

The Problem:

Historically, children with handicaps have been excluded from public school programs. In 1919, the Wisconsin Supreme Court excluded a child with cerebral palsy (Beattie V. State Board of Education) because of his "depressing and nauseating effect on the teachers and school children and... (because) he required an undue portion of the teacher's time" (Martin, 1980). The philosophy of the day was to exclude handicapped children if they were deemed incapable of profiting from public education. Teacher training programs were, and to a certain extent still are, geared toward teaching an adopted text-based curriculum to "normal" children. Children not able to respond to this curriculum are often viewed by the teacher as a hindrance to the rest of the class.

These "difficult to teach" children were often segregated in state-supported institutions but were not allowed to take part in a regular school education. This exclusion policy existed in all states although states would sometimes provide support money or services for certain handicapping conditions. For example, until 1971, mentally retarded children were denied admission to public school programs. In a landmark case, the Pennsylvania Association for Retarded Children v. Commonwealth of Pennsylvania won for the retarded, not only access to public school programs, but also tuition and maintenance costs in approved institutions and home bound instruction where appropriate.

The P.A.R.C. case was shortly followed by the case of Mills v. Board of Education. The practices under attack in the Mills case involved all types of handicapped children either denied access to schools, suspended or expelled if the schools did not want to serve them.

Following these two cases, 36 other right-to-education cases were soon filed and the proverbial handwriting was on the public school house wall. The federal government decided it was time to establish a clear standard.

This lack of funding has raised the following havoc in many districts:

* Due to tight local budgets, districts were unable to hire the additional staff necessary to handle the larger number of students receiving special education services and clinicians found their caseloads overwhelming.

* Special education staff ended up serving a wider variety of handicapping conditions.

* In order to provide an "appropriate" education, clinicians found themselves needing to prepare more curricular areas.

* Clinicians found increasing amounts of time being spent in report writing, testing and in meetings associated with communication about identified children.

The end result was, clinicians found themselves either seeing children less often or seeing children in larger groups. This decreased the effectiveness of their programs and their job satisfaction.

Proposing a Solution:

Currently, microcomputer technology has the potential to assist special educators in three major areas:

1. computerized testing (administration and/or scoring);
2. computer assisted instruction (using the computer as a tool and also for extra repetition in skill development);
3. computer managed instruction (record keeping and report generation).

The following is a brief description of what is available in each of the areas.

Computerized Testing and Scoring

There are currently four or five companies that nationally advertise computerized test administration and scoring. The price of these packages range from $500 to $20,000 and are designed to run either on large timeshared systems or eight bit systems such as an Apple or T-RS-80. Some of the less expensive systems require the clinician to key into the computer the test answers, rather than by using a card reader to automatically read the test results into the computer.

At this time the systems designed to both administer and score tests seem to be aimed toward psychological testing. However, many companies are engaged in the development of software designed to score both psychological and educational tests given by an examiner. (Some of these packages also generate reports.) Companies claim that the test results can be entered into the computer by a secretary or an aide and in most cases, take no longer than 10 minutes to key in and obtain a complete set of scores. The cost of these scoring systems (excluding those that also generate reports) seem to be in the $300 to $500 range.

Although other laws were enacted the one finally tying compliance of federal guidelines to reimbursement monies was the famous Education of all Handicapped Children Act: P.L. 94-142, passed in 1975.

At that time, of the approximately eight million handicapped children requiring special education and related services, the Bureau of Education for the Handicapped estimated that only half were receiving an appropriate education. Congress, appalled by this lack of service, mandated a search for children not being served and called it "Child Find." "Child Find" was <u>mandated</u> to:

a. find and serve all children between the ages of 0 and 21 if they qualified under one of the 11 handicapping conditions listed in P.L. 94-142.

b. provide additional services to children already identified if deemed necessary in order to provide an appropriate education by a child study team.

A larger percentage of the public school population were referred, evaluated and placed on Individual Educational Plans. In 1976 and 1977 as awareness grew, special education departments nationwide began scrambling to adopt service delivery models that would bring them into compliance and thus eligible for federal funding of programs.

As a result the following happened:

* Children formerly served in institutions outside the community returned home expecting to receive a "free and appropriate education in neighborhood schools."

* right-to-education cases became increasingly common.

* special education departments expanded their services to be able to provide for the 11 handicapping conditions described by P.L. 94-142. (These conditions include: deaf-blind, hard of hearing, mentally retarded, multihandicapped, orthopedically impaired, other health impaired, seriously emotionally disturbed, specific learning disability, speech impaired and visually handicapped.)

* Caseloads burgeoned as identified children began being properly identified served.

Never was the public more aware or sympathetic to the needs of the handicapped; never were parents more willing to go to battle to protect the educational rights of their handicapped children. Things had never looked more promising until the reality that the proposed funding levels attached to the passage of P.L. 94-142 did not materialize. Currently each school district receives about $175.00 for each child having an IEP (not to exceed 12% of their school population). [illegible]

Computer Assisted Instruction

This is the area currently receiving the most attention from clinicians and the area where most microcomputer software is available. In the booklet entitled, Learning Disabled Students and Computers: A Teacher's Guide Book, Metzger, Ouellette and Thormann posit.

"Microcomputers can assist your efforts to meet the needs of learning disabled children by:

* Helping meet the individual needs of children as they acquire skills in reading, writing, arithmetic and problem solving.

* Allowing the child to interact with a well-prepared tutor which provides immediate feedback.

* Serving as a personal tool in the completion of routine academic tasks.

* Providing the child with an environment in which he/she may take charge of the machine, commanding it to perform some activity of interest to the child.

* Providing virtually unlimited time and patience in serving students and teachers as they work toward meeting the goals of each child's IEP. (Metzger, Ouellette, Thormann 1983)

One can imagine that the clinician would be quick to embrace a technology that would allow them to reach the goal of helping each handicapped child to become more successful in the school setting. E. Paul Goldenberg says it beautifully in his book, Special Technology for Special Children.

> "It is not their special needs with which I am primarily concerned here, but their normal needs. To have an enjoyable and esteemable life, to be able to interact satisfyingly with one's environment and its people, its things and demands; these are the needs of which I speak. Sometimes we require special techniques and technology to help us meet our needs."

The major findings coming from current research on computer assisted instruction revolve around time-on-task. Children seem to be acquiring more skills simply because the format is so motivating and non-threatening that they are attending to the instruction for longer periods of time. CAI is not magic but educators of special children have always realized that it's not that the children can't learn, it's that an effective way to teach them has not been found. They also realize that getting and keeping such a child's attention is often difficult. Perhaps the format offered by computer instruction will help alleviate both of these problems.

Computer Managed Instruction

Turning the computer into an IEP writing and management machine is a clinician's dream. The record keeping associated with P.L. 94-142 is overwhelming to people who would rather spend their time working with students.

Some of the report writing programs are "shells." The person writing the report selects key phrases to be inserted and a data entry person enters the information and generates the printed copy. Other programs offer the clinician large sets of goals and objectives to choose from by using a check list. Many of the programs claim to cut IEP planning time in half (some say "significantly reduce") and promise "personalized IEP's available in minutes." The more expensive systems allow the IEP planner to specify and/or enter: a) goals, b) objectives, c) assessment data d) related services, e) placement, and f) classification. Most of these systems are designed specifically for public school use and operate on both a timeshared computer and a microcomputer. The cost for such a software package will be in the neighborhood of $1,000.00.

Conclusions:

If departments of special education are to avoid a tremendous public backlash against their programs they must act quickly; indeed, it may be too late. Already the public, not understanding the problem, is beginning to decry the large sums of money being put into special programs.

Beverly L. Anderson, Director of the National Assessment of Educational Progress, reporting for the L.A. Times in an article titled "Education/Ignoring the Top," states that our nation's schools are committing "unthinking, unilateral educational disarmament." Her conclusion:

"We have arrived at this sorry state over the past two decades in part because we have concentrated so much of our efforts at the lower end of the school ladder, in raising the achievement levels of disadvantaged students, that we have neglected our most able students and the critical skills they need to master."

The cry is being heard for academic excellence, programs for the gifted and for schools to enter the era of "electronic learning." Expenditures will be directed towards these academically able students, not towards the handicapped.

Seymour Papert's now famous quote, "If a school only has two computers, then look for a small pocket of students - perhaps a class of learning disabled youths - and give them the computers," gives special educators their lead. The reality is, no school board is going to magically inflate budgeted amounts for the handicapped. The model of service delivery must change and computers have the capacity to help affect that change.

It would seem remiss not to end with a word of caution. The types of software packages described in this paper must be carefully evaluated by persons knowledgeable in the field of special education. The software must be capable of assisting clinicians and students in a way that is educationally sound and only thorough evaluation of systems prior to purchase will allow that determination.

REFERENCES

Goldenberg, E. Paul, Special Technology for Special Children: Computers to Serve Communication and Autonomy in the Education of Handicapped Children. Thesis presented to Harvard University, 1977, 192 pages.

Martin, Reed, Educating Handicapped Children, The Legal Mandate. Illinois: Research Press Company, 1980. 181 pages.

Metzger M., Ouellete, D., Thormann, J. Learning Disabled Students and Computers: A Teacher's Guidebook. Eugene: International Council for Computers in Education. 1983. 48 pages.

PROBLEMS IN COMPUTER GRAPHICS

Dr. Lee T. Hall
Southern Oregon State College

The topic "COMPUTER GRAPHICS" covers a broad area of computer application. I have seen the term applied to bar graphs, pie graphs, design of custom characters, vector drawings, 'sprites', and a host of other visual presentations. When we look at hardware involved in computer graphics, we may have to consider plotters, tablets, cathode ray tubes (CRT's), plasma panels and even printers. In the case of CRT's, we have the choice of direct storage tubes, refreshed tubes, random scans and raster scans. When one looks at a new computer and asks the question, 'What are the graphics capabilities?', the answer could involve graphics 'pages', bit maps, shape tables, color pages, text pages, graphics characters and an array of BASIC functions and POKEs.

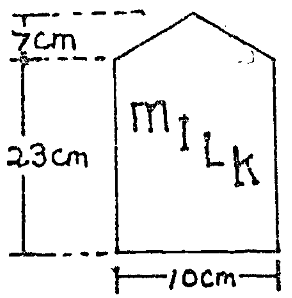
In order to narrow the topic of computer graphics to manageable proportions, I will limit my discussion to that of microcomputers with raster scan CRT's (ordinary television monitors). In particular, I will limit the discussion to the vector drawing capabilities found on the Apple II+, the ATARI 800, and other computers with similar graphics pages (such as the Commodore 64). These machines have many other graphics or graphics-like capabilities in addition to vector drawing. My talk will center on the problems of clipping, scaling, and windowing in two dimensions. I will also present the problem of viewing a three dimensional scene and displaying it in two dimensions.

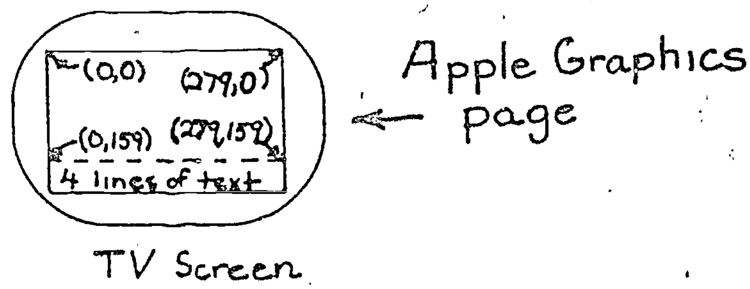
Scaling and Windowing

Any object that one might wish to display with the aid of computer graphics can be described in mathematical term using the actual dimensions in numbers of feet, meters, inches, miles, etc.. The choice of the unit should not effect the display of the object using computer graphics. Since the numbers we are referring to represent the dimensions of the object, they will always be positive. However, since we could be called upon to display something as small as the orbit of an electron or as large as the solar system, the numbers of these units can vary greatly. In each of the graphics systems that we are discussing, the range of numbers we may use to address individual pixels in the graphics pages are very limited indeed. The problem of converting the numbers describing the actual dimensions of the object into numbers that can address points or pixels on the graphics page is called scaling.

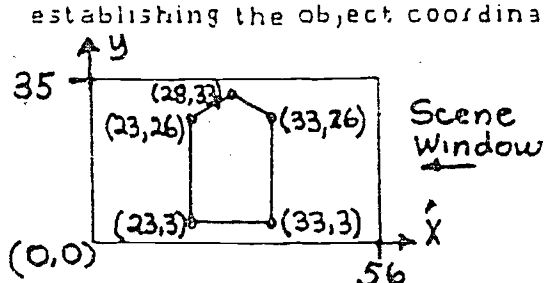
Usually when we display an object, we refer to it as 'the scene'. This recognizes the fact that in order to view an object, we may have to present certain other related objects as well. A house would look somewhat strange without the foundation or yard upon which it rests. The scene containing the object that we are viewing generally does not have any natural borders. For example, if we view a buiding, how much of the surrounding countryside should be included in the scene.. only the building lot, a city block, the rest of the city? On the other hand, the graphics pages used by the computers have very definite boundaries. The designation of artificial borders for the scene that we wish to display is called windowing. These borders allow us to state a relationship between distances in the scene and coordinates for the graphics page.

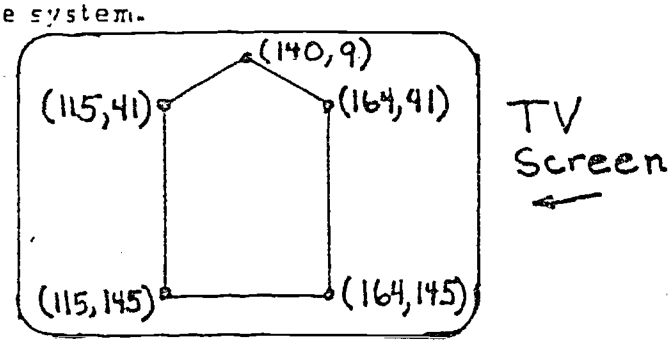
Consider the following example: We are viewing the side of a milk carton which stands 30 cm high and has a width of 10 cm. We will display this scene on an Apple high res graphics page which is 160 by 280 pixels.





We choose a window around the milk carton that will permit us to see the entire object (the milk carton). We also take care to choose a window that has dimensions somewhat proportional to the viewing screen. This requires that the height to width ratio be approximately 5 to 8. (If we are using full page graphics then 6 to 8 would be appropriate). We will center a 35 cm by 56 cm window on the milk carton. A 35 by 35 window would distort the view by making the carton appear wider than it actually is when displayed on the TV. We could eliminate this distortion by setting a second window in the graphics page that would correspond to the one just set in the scene. I will discuss this approach later on. We are now ready to define an 'object coordinate system'. An object coordinate system describes the scene in terms of its natural dimensions. In doing this we usually assume that the origin (pt. 0,0) of the object coordinate system is in the lower left-hand corner of the window and that the X,Y coordinates are positive real numbers throughout the window. A better choice for the origin would be the upper left-hand corner but then the resulting coordinate system would be non-standard. Nothing would be lost with this orientation, but old habits die hard. We will continue with the standard method of establishing the object coordinatate system.





Let (X,Y) be the object coordinates for a point in the scene. The point in the graphics page that correspondes to the point (X,Y) has coordinates (SX,SY) which are given as follows:

SX = 279*X/56 and SY = 159-159*Y/35

We call these the scaling formulas. You must set a window before they can be created. The 279 and 159 are both one less than the X and Y dimensions respectively of the Apple graphics page. The 56 and 35 are the dimensions of the window that we just set.

To draw the scene we would convert the object coordinates of the corners of the milk carton into screen coordinates and then connect them using HPLOTs. It is very important that we draw only between appropriate corners of the scene. In order that this is done properly, we use subscripted variables and make them correspond to the order in which the vertices should be connected. The following program segment accomplishes this result:

```
10 DATA 23,3
20 DATA 33,3
30 DATA 33,26
40 DATA 28,33
50 DATA 23,26
60 DIM X(5),Y(5),SX(5),SY(5)
70 FOR J=1 TO 5:READ X(J),Y(J):NEXT J
80 FOR J = 1 TO 5
90 SX(J)=INT(279*X(J)/56+.5)
100 SY(J)=INT(159.5-159*Y(J)/35)
110 NEXT J
```

Lines 90 and 100 are the scaling formulas required for this scene. The INT function correctly rounds the real numbers that are produced by the scaling formulas when we add .5. The use of INT is not necessary but makes the result more accurate.

To draw the scene, we simply connect the screen coordinates in order. The following program segment does this:

```
200 HPLOT SX(5),SY(5)
210 FOR J = 1 TO 5
220 HPLOT TO SX(J),SY(J)
230 NEXT J
```

There are many ways that the above procedure could be streamlined. One way would be through the use of two-dimensional arrays. This would eliminate the need for both the X and the Y arrays. Such improvements are not included in this discussion since they tend to obscure the underlying principles.

Viewports

A more general approach to windowing and scaling requires that we set a window in both the scene and the graphics page. The windowing technique presented above only allows us to display scenes on the entire screen. This is somewhat limiting if we have more than one scene to display or if we wish to present text alongside the scene. The window in the scene is set as before. A window in the graphics page (i.e. on the TV screen) is called a viewport. Think of a viewport as a window in our TV screen through which we can see the scene. We set a viewport by designating the screen coordinates of the left edge, the right edge, the top, and the bottom of the window. In order to do this, we will define the following variables in our program:

LM is the left margin of the viewport.
RM is the right margin.
BM is the bottom margin.
TM is the top margin.
XR is the width of the window set around the scene.
YR is the height of the window set around the scene.
(In the previous example XR was 56 and YR 35.)

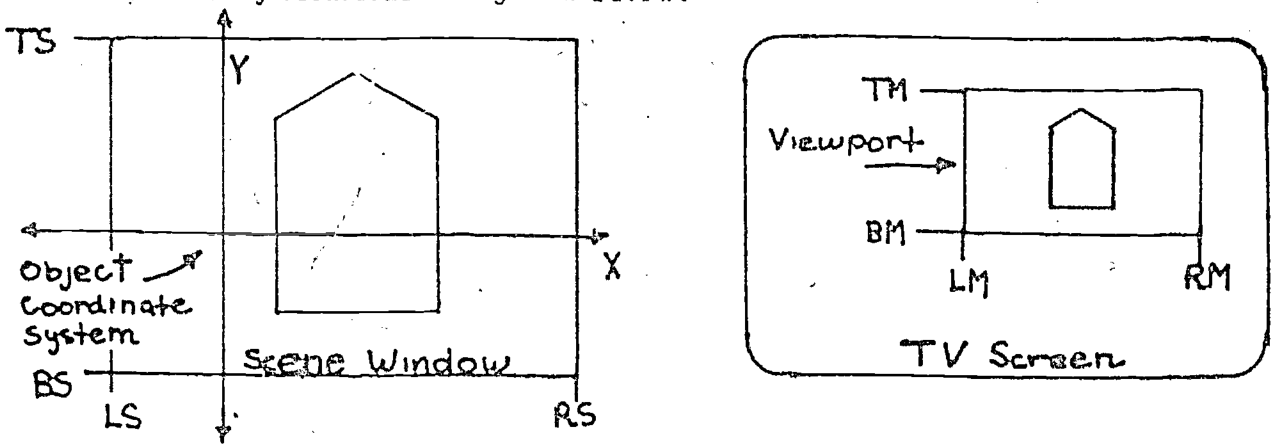
Of course, we no longer need to maintain the ratio 8 to 5 for the width to the height of our window. In place of this we must attempt to make the ratio of width to height the same for both the viewport and the scene window. Unfortunately, 10 pixels measured horizontally is not the same distance on the TV screen as is 10 pixels measured vertically. I suggest that you experiment with different window settings to achieve the most realistic results. Using the above variables in our program, the scaling formulas become:

```
SX = (RM-LM)*X/XR + LM          (s1)
SY = BM - (BM-TM)*Y/YR          (s2)
```

We may generalize this process even further by not requiring that the origin of the object coordinate system be located in the lower left-hand corner of the scene window. This is particularly useful if there is a natural pre-existing coordinate system that does not correspond to the window we wish to set. For example, we want to draw a map of a parcel of land by using the surveyed property line description. The coordinate system used in this case has its origin at the equator on the Greenwich Meridian. There are many other less extreme examples where we might prefer to have the object coordinate system independent of the window around the scene. In this case we must designate the boundaries of the window containing the scene in the same manner that we did for the viewport in the graphics page. We define the following variables in our program:

LS is the object coordinate of left side of
the scene window.
RS is the object coordinate of right side.
BS is the object coordinate of bottom.
TS is the object coordinate of top.

If we wish to view a scene in the window and viewport indicated, the necessary scaling formulas are given below.



SX = (RM-LM)*(X-LS)/(RS-LS) + LM (s3)
SY = BM - (BM-TM)*(Y-BS)/(TS-BS) (s4)

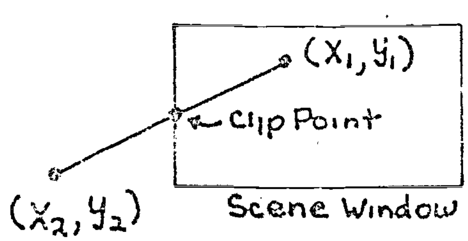
The XR and YR values in the previous formulas are replaced by (RS-LS) and (TS-BS) respectively. If you set BM = 159, TM= 0, LM = 0, and RM = 279, the above formulas become:

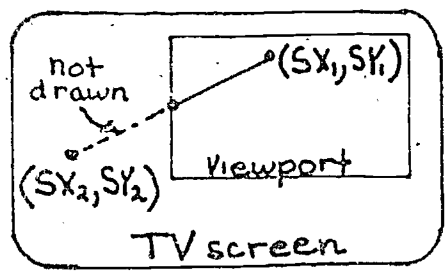
SX = 279*(X-LS)/(RS-LS) (s5)
SY = 159 - 159*(Y-BS)/(TS-BS) (s6)

These formulas display the scene within the scene window on the entire TV screen. If you are using an ATARI or any other computer, formulas s1 ... s4 apply as they are written. For the ATARI or Commodore 64, change formula s5 by changing the 279 to 319. Formula s6 applies as is.

Clipping

Once we have established the window(s) and the scaling formulas, another problem is created. That problem results when we try to draw to a point that is outside of the scene window. This can be avoided if we very carefully set our scene window around all objects that we intend to draw in the graphics page. However, if there is animation involved, the object that we are displaying may move out of the window. Or, if the object is part of some larger object which extends outside of the scene window, we may have to draw that portion of the larger object that appears in the window. (We may even wish to 'zoom in' on some small portion of the object we are displaying--thus forcing part of it (the object) outside of the scene window. In any case, we most likely will have to draw to some points outside of the window. We may convert the object coordinates of these points to screen coordinates by using the scaling formulas. (The resulting screen coordinates will not address pixels in the viewport if the original point was outside of the window). When we try to draw to these points in the graphics page, we will discover that either our program aborts (RANGE ERROR) or we leave the viewport set earlier. In either case, the result is undesirable. The process of avoiding this problem is called clipping. Clipping has two parts: A procedure to determine when a point is outside the scene window; a procedure to clip any line drawn from a point within the viewport to a point outside of the viewport at the place where that line intersects the viewport boundary. This point on the edge of the viewport is called the 'clip point'.





The first part of the problem is usually handled by comparing the object coordinates of the points in the scene to LS, RS, BS, and TS. By locating the point of intersection of the window boundary with the line on the object that we are displaying, we have also located the clip point as well. The clip point will be the screen coordinates of this point. This fact is used in the second part of the clipping process when we search for the clip point. For example, suppose we wish to draw a line from (X1,Y1) to (X2,Y2) but X2 < LS (i.e. (X2,Y2) is to the left of the window). The following formula would locate the Y coordinate of the point where the line drawn between (X1,Y1) and (X2,Y2) crosses the left window boundary:

```
Y = Y2 + (LS-X2)/(X1-X2)*(Y1-Y2)        (s7)
```

The coordinate of the point on the left window boundary would be (LS,Y). The LS is the x-coordinate of all points on the left window boundary. These coordinates would now replace the old point (X2,Y2) provided that the newly computed Y2 coordinate placed the point on the scene window boundary. (It is possible that the new point, (X2,Y2), would still be above or below the actual window). Suppose that it is below the window. In this case Y2 < BS, and we compute the new X2 coordinate as follows:

```
X = X2 + (BS-Y2)/(Y1-Y2)*(X1-X2)        (s8)
```

The coordinates of the point on the boundary would be (X,BS). Only one of the above formulas need be applied if the first application yields a point on the window boundary. There are two other formulas which are used in the clipping process. These are needed when X2>RS or Y2>TS and they are:

$$Y = Y2 + (RS-X2)/(X1-X2)*(Y1-Y2) \qquad (s9)$$
$$X = X2 + (TS-Y2)/(Y1-Y2)*(X1-X2) \qquad (s10)$$

The formulas s8 and s10 can not be applied when Y1=Y2 (i.e. the line is horizontal). In this case, the intersection is easily found by taking LS or RS, whichever is appropriate, to be the X coordinate. Similarly, the formulas s7 and s9 can not be applied when X1=X2 (vertical line). The formulas s7 ... s10 are sufficient to handle most clipping problems, but the decision of which formulas to use and when more than one is required is a complicated process. An excellent routine for clipping can be found 'MICROCOMPUTER GRAPHICS' by Roy E. Myers, pub.: Addison Wesley, pg. 103. A more general clipping procedure called the Cohen-Sutherland algorithm can be found in many graphics texts. It is usually better to avoid clipping whenever possible.

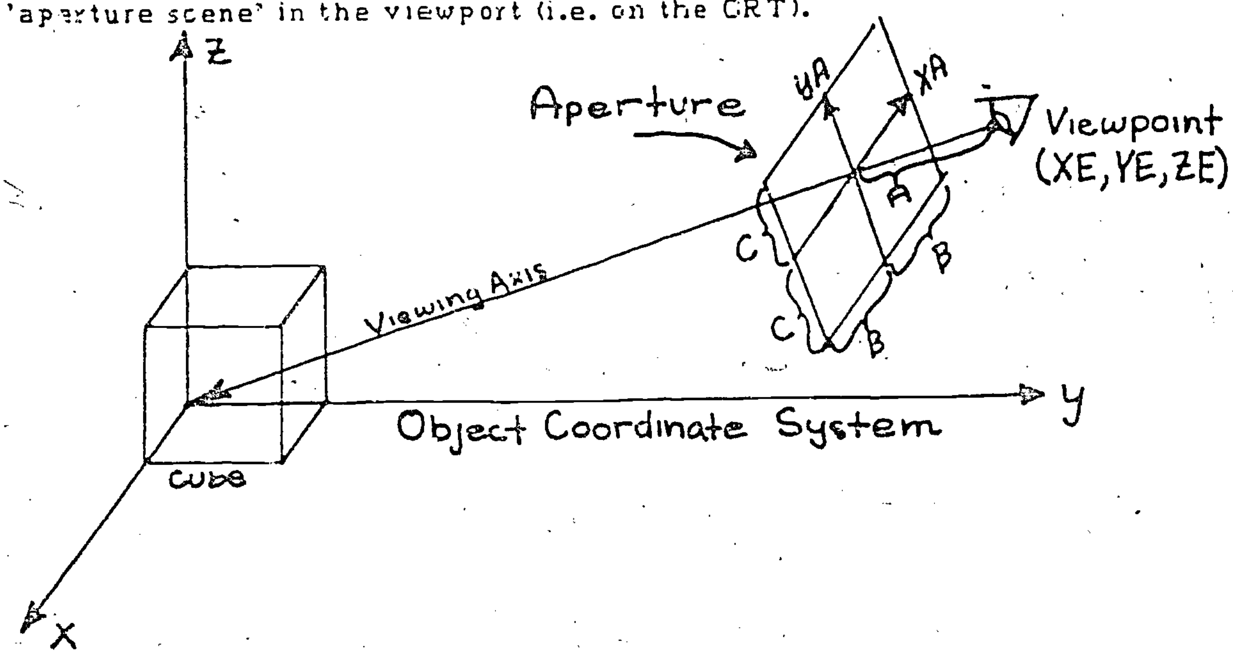
Three-Dimensional Viewing

Thus far we have concentrated on veiwing a two-dimensional scene. The scenes that we can view in this manner are somewhat limited. If we wish to view 'real' life, we soon run into another dimension. That is to say, we discover that part of the scene is further away from the viewingpoint than are other parts. In order to realistically display a three-dimensional object on the two-dimensional TV screen, we add what is called perspective. To understand perspective, imagine a yardstick being held at an arm's length. Whether it is held horizontally or vertically, its length extends across most of what we can see. If a friend took the same yardstick and held it 20 feet from us, it would only extend across a small portion of what we could see. If this same friend got in a car and drove to the other side of town with the yardstick, the friend and the yardstick may appear as a single point in our view (if they are seen at all). Perspective takes into account that equal distances perpendicular to the viewing axis (the direction we are looking) subtend smaller viewing angles the further they are from the viewpoint.

The problem of displaying a three-dimensional object on a two-dimensional TV screen (or viewport) is one of projecting three dimensions onto a two-dimensional plane, taking perspective into account. To develop a comprehensive solution to this problem would certainly take more space than I have here. I can, however, relate the terminology of the problem and provide you with the two equations that create a viewing transformation based on the location of the viewpoint and the origin of the object coordinate system. A viewing transformation converts the coordinates that describe the three dimensional scene (object coordinates) into screen coordinates that would allow you to display the scene on the CRT. The perspective would give the impression that the scene had three dimensions. The following terms are used in the solution of this problem and are illustrated in the figure below.

Object Coordinates: The coordinate system is used to describe the three-dimensional scene.
Viewpoint: The location, in terms of object coordinates, from which the scene is viewed.
Viewing Axis: A line drawn from the viewpoint to the origin (usually) of the object coordinate system.
Aperture: A rectangular window perpendicular to the viewing axis through which the scene is viewed. This corresponds to the scene window in the previous discusion of two dimensional scenes. It should be noted that adjusting the aperture can create both wide-angle and telephoto effects in our scene.
Viewing Pyramid: This is created by drawing lines from the viewpoint through each of the four corners of the aperture. Anything inside the pyramid is in the scene and anything outside the pyramid is not seen.

These terms are illustrated below. The object being viewed is a cube. Notice that the XA,YA coordinate system is superimposed on the aperture. This coordinate system cooresponds to the object coordinate system in the display of two-dimensional scenes as discussed in the first part of this paper. The process of viewing a three-dimesional scene can be divided into two parts. First, project the three-dimensional scene with prespective onto the aperture, and second, display this two-dimensional 'aperture scene' in the viewport (i.e. on the CRT).



The process of creating (XS,YS) coordinates from (X,Y,Z) object coordinates is complicated indeed. As previously indicated, the process is known as a viewing transformation. This transformation depends on just four things:

1) The coordinates of the viewpoint.
2) The distance from the viewpoint to the aperture.
3) The width of the aperture.
4) The height of the aperture.

Let the coordinates of the viewpoint be (EX,EY,EZ) and the aperture be A units from the veiwpoint with a width of 2B units and a height of 2C units. The following formulas give the viewing transformation:

$$XA = \frac{A*T*(Y*EX - X*EY)}{V*(T*T - X*EX - Y*EY - Z*EZ)} \qquad (s11)$$

$$YA = \frac{A*(V*V*Z - EX*EZ*X - EY*EZ*Y)}{V*(T*T - X*EX - Y*EY - Z*EZ)} \qquad (s12)$$

In the formulas above (s11 and s12), the values V and T are defined as follows:

V = SQR(EX*EX + EY*EY)

T = SQR(EX*EX + EY*EY + EZ*EZ)

These formulas assume that the viewing axis is pointed towards the origin (0,0,0) of the object coordinate system. To convert the XA,YA coordinates to screen coordinates, one must use the scaling formulas developed in the first section of this article. For example, to display the aperture on the entire Apple screen, the following formulas would do:

XS = 279*(XA + B)/(2*B) (s13)

YS = 159 - 159*(YA + C)/(2*C) (s14)

Summary

Creating a graphics display requires that one convert the natural units which describe the scene into coordinates that address points (or pixels) on the graphics page. In two dimensions, this is called clipping, windowing and scaling. In three dimensions, one must first project the scene onto an aperture which is in two dimensions and then

96

copy the aperture to the graphics page. Most of the mathematics involved in clipping, windowing and scaling is simple proportion. / The creation of a viewing transformation which changes coordinates from three dimensions into screen coordinates in two dimensions is done largely by matrix representation and matrix multiplication. The formulas given in this paper, though limited to the application indicated, should serve to handle most two- and three-dimensional scenes.

Suggested Readings

Myers, Roy E. 'Microcomputer Graphics'. Addison Wesley, 1982. 281 pages.

Rogers, David F. & Adams, J. Alan 'Mathematical Elements for Computer Graphics'. McGraw Hill, 1976. 237 pages.

Newman, William M. & Sproul, Robert F. 'Principles of Interactive Computer Graphics'. McGraw Hill, 1973. 607 pages.

COMPUTER ASSISTED INSTRUCTION AND GRAPHICS

Beverly J. Jones

Introduction

This presentation will demonstrate Apple Display Utility and some Graphic Utilities developed as a result of a recent study described below.

Purpose:

The purpose of this study was to design and construct integrated components of computer software and hardware which would facilitate the use of computers by art educators for instruction and as a graphic medium.

Procedure:

Prior to designing and constructing the final products the following research was conducted:

1. Literature search for instances of computers in instruction in the arts and humanities and as a graphic medium. Both hand searches and computer searches of large data bases were conducted.
2. A questionnaire was mailed internationally to individuals identified in the literature search and from participation in conferences. Further correspondence, phone conversations and exchange of research papers with some of these individuals provided additional information on "cutting edge" use of computers in the arts, especially those uses which have educational implications.
3. Informal interviews were conducted with public school teachers and students who were successfully using computers in a variety of settings. Interviews were also conducted with some teachers who did not use computers and with those expressing antipathy toward computers.
4. An examination and review of many examples of educational and graphic computer software programs and an examination of computer hardware peripheral devices was conducted.
5. Current graduate level courses in computer science and physics were audited as an aid to assessing immediate feasibility of theoretical concepts I was developing and to achieve awareness of future possibilities.
6. All of the above information was considered in the light of selected research and curricular theory in general education and art education.

Based upon the information gathered and assimilated in the previous steps a conceptual model for the final products was designed. Among the most crucial design decisions were those which indicated the desirability of making the programs and hardware easily used by the novice and transparent enough for those wishing to understand the programming and electronics involved. The decision to allow departure from instructional programming as an "expert controlled" experience was crucial. For example, these decisions led to the inclusion of economical homebuilt hardware and to the use of BASIC as the primary programming language.

The next steps involved the design of the components which may be used together in an integrated instruction package or use separately. Finally these components were constructed. This required the development of complex sophisticated programs which were then reconstructed for novice users at the cost of some sophistication and operating speed. To aid me in the final stages I hired a microcomputer consultant and three students. The consultant, Bill Crouch, programmed the instructional components, Apple Display Utility and Guided Tour. His son, Brian Crouch (South Eugene High School), programmed the advanced graphics. Mark Peting (South Eugene High School) and Steve Jazdzewski (Medford High School) constructed the slide projector prototypes.

Outline and Description

The final product for the Apple Education Foundation consists of two integrated components. These integrated components are unique in the manner in which they provide flexible instructional programming for computer assisted instruction.

I. Software

A. Two programs for use in constructing educational or commercial group demonstrations or individualized instruction. These are titled:

1. Apple Display Utility
2. Guided Tour

B. A set of advanced graphic utilities to be used independently or with the programs in 1. These are titled:

1. Brush
2. Line
3. Block
4. Paint Set
5. Graphics Program #1

II. Computer Hardware

A. A controlling interface between the computer and two carousel slide projectors simple enough to be built in a high school electronics laboratory. Schematics are included. This is used in the programs, Apple Display Utility and Guided Tour. This hardware adds access to photographic images simply and at low cost so they may be integrated with the small text, large text and graphics already available from these two programs.

Further Description of Each Component

I. Software

A.1 Instructional and Demonstration Software: Apple Display Utility

Description: This program allows a person with no background in computer programming to create individualized instruction or displays for group demonstration using a text editor (Apple Writer or Apple Pie). A one or two disk drive system may be used.

By typing single letter commands the user may invoke standard size text, large high resolution text easily seen from the back of a classroom, display graphics pages, advance the slides in two carousel projectors, or run subsidiary programs and return to the correct place in the main program. Programs may be Exec files, machine language, Applesoft or Integer Basic. Thus, for example, a teacher may project slides and accompany them with a video display of small or large text or with a diagram or dynamic graphic simulation in low or high resolution. Simple commands also allow pauses, page shifts and other format considerations. Although this program does not preclude the use of sub programs which are multiple choice or other standard programmed instructions it is designed to encourage maximum flexibility of instructional mode and materials and to utilize the unique characteristics of the Apple computer.

A.2 Instructional and Demonstration Software: Guided Tour

Description: Guided Tour has all the features of Apple Display Utility with the following exceptions. It does not allow the user to enter commands via a text editor and thus requires some knowledge of computer programming. Users must be able to renumber and insert correctly any commands the sub programs in appropriate positions and with correct programming transitions. However, it runs much faster than Apple Display Utility and has some extra features. It has a built in subroutine called "Names" which records students' names and their progress through the lesson. It allows students the option of returning to their last position in the lesson or selecting another entry point. It also has features which allow the programmer to select 80 columr display or use a Votrex chip to add "computer speech" to the lesson or demonstration. These features are included to illustrate its flexibility to interface with other peripherals. The inclusion of seven slot options for this peripheral selection make this program quite flexible for instruction and display.

B. Graphics Software: Five Programs

Description: Graphics Utilities has five sections, each of which provides the user specific high resolution graphic techniques. The sections are compatible with one another. This allows a single image to be built up using the techniques from several of these programs. The sections are Brush, Line, Block, Paint Set and Graphics Program #1. A brief review of commands and procedures will reveal the variety of these techniques. These include simple line graphics with rubber banding, move, rotate, scale; brush graphics with size and shape of cursor defined by user, fills--either solid, color mix or crosshatch with size of crosshatch and whether lines or holes specified; block graphics allows encloding and display of a section of a previously created shape, windowing, clipping, repeated printing. This section utilizes a machine language subroutine called Graphics-Or. This subroutine permits previously created graphics or sections of graphics to be stored in memory on graphics page one and graphics page two and then merged. The merge option follows Boolean logic and allows and, or, exclusive-or.

Graphics Program #1 has the additional feature of allowing a user to specify whether each shape as it is created should be in front of or behind the picture plane. It also has a fill which allows the user to select the increment of the fan which fills the shape and to select if the fan is made of dots or lines. This incremental fan fill makes it possible for shapes specified as behind another shape to exhibit a transparent effect in the overlap between them and the foreground shape.

Together these programs should prove helpful to students in graphic arts and design to construct many slightly varied examples rapidly. Teachers will also find these programs useful in constructing visual aids for instructional purposes.

II. Hardware

Slide Projector Controllers

Description: These use elementary electronic circuits and parts readily available from neighborhood electronics stores. It is hoped that the construction of these may dispel the "black box" fear of electronics that some students and teachers have. Because they are so simple and intended to be built as instructional experiences they are not very sophisticated in use. They do, however, provide access to two projectors of photographic images to incorporate with the instructional and demonstration programs in this package. The only commands used by these are advance slide projector one or advance slide projector two. One of the points intended to be made by the inclusion of this simple hardware piece is that with even minimal knowledge of electronics we do not need to wait for large amounts of funding or resources or for developments in high technology such as video disk to become commonplace. Rather, we can devise our own means to achieve selected ends, in this case to incorporate photographic images such as painting, sculpture and architecture in our computer assisted instruction.

MICROCOMPUTERS IN SCHOOL SCIENCE

Dr. Gene A. Stringer
Alpine Educational Computing
and
Southern Oregon State College

How would you like to have an assistant in your classroom that always did precisely what you told him to do? Suppose he had the memory of an elephant, could recall information in a flash and communicate it in written, graphical or spoken form. He would work all day and all night if necessary, never demand a break or a raise in pay. In fact he would work for practically nothing and often with no supervision. All you really need to do is hand him the necessary tools and stand back. I suppose I should mention one small handicap. He doesn't speak English.

Of course we all realize I have been talking about the ubiquitous computer. I have taken this means of introduction to emphasize the capabilities we so often take for granted in this increasingly useful machine. Before discussing the applications that are now emerging for computers in the science classroom I will take a few minutes to review the general characteristics of educational software packages that help to improve their usefulness.

Useful System Characteristics

User Friendly Software

Most computer users will never learn the native tongue of the computer known as machine language. Fortunately, there are a few computer enthusiasts around who obtain a great deal of satisfaction, both intellectual and monetary, from bridging the gap between machine language and user friendly software. With clever programming they have taught the computer to communicate to us in English. Among the important characteristics of user friendly software are the following;

1. No programming skills are required of the user,
2. Any user input produces a reasonable response,
3. The user does not need to decipher a manual.

Currently, most programs leave a lot to be desired in this area but the situation is gradually improving.

Intelligent Use Of The Resource

This refers to the question of whether the application makes significant use of the computer resources that we have available. All too often the application could just as well have been presented in a textbook. The implication and promise of the computer in the classroom is that it provides greater flexability and learning efficiency than a textbook. What are some of the characteristics we should expect in a well designed computer application?

1. The system is interactive,
2. Avoids unnecessary repetition,
3. Is able to recognize a wide variety of user inputs and delivers an appropriate response,
4. Is capable of generating a new variation on the theme each time it is used,
5. Makes good use of graphical forms of communication,
6. Uses available input/output devices in addition to the keyboard.

As you examine some of the software/hardware systems that are now available for science education look for these charcteristics. They will help enhance the long term usefulness of the package.

Classifying Computer Applications

Any classification system has a degree of arbitrariness about it but most computer applications will fall into one or more of these areas; games, drill and practice, tutorials, simulations, instrumentation, utilities, development tools or programming languages. You may have a personal preference for certain types of applications but remember it is possible to effectively use or misuse any type of software. The choice of software is best left to the individual instructor. Then the chances for successful implementation are much greater.

Experience has shown that the first applications that are usually implemented on a new computer system are games. They can be great motivators and often have sound educational objectives. Drill and practice is important for developing basic skills that require routine, repetitive work with prompt feedback. A good example of this is the Little Professor sold by Texas Instruments which was more or less successful in helping my kids to learn basic skills in arithmetic. A more sophisticated example of drill and practice is Control Data Corporation's new implementation of the PLATO learning materials. The physics package contains several interesting features. The problems do not give the user any data to use in the computation. Instead it allows the user to request different kinds of information that they believe will be useful to the solution of the problem. This encourages a wide variety of possible approaches to the solution of each problem. In addition, the numerical values given to the user are varied each time the software is used.

Tutorials are very useful for individual student learning or for presentation of material during a lecture/lab situation. For example, a recent laboratory session began with the Optics tutorial from Cross Educational Software. It was used to introduce the topic of ray diagrams and image formation. Later during the lab, students were able to individually review the tutorial where they had questions.

Simulations provide a great opportunity for interaction with special kinds of peripheral devices such as game paddles and joysticks. It is perhaps unfortunate that the best examples of simulation are in video games. If manufactures of game products would put as much effort into educational software as went into Choplifter or Castle Wolfenstein we would have some really astounding simulations to use in education. One caution about simulations of real world situations-the computer can be programmed to produce any kind of output. Let the user beware. It may or may not be a very close approximation to the real world. On the other hand simulations can be used to illuatrate effects that normally could not be observed in any other way. A good example of this is the program Rendezvous, created by NASA's Jet Propulaion Laboratory, which simulates the launch and docking of a apace craft.

Instrumentation packages are special hardware and/or software tools that allow the computer to interact directly with laboratory experiments. The computer is then able to perform such varied tasks as the measurement of time, temperature, voltage, light intensity or you name it. There are many different techniques for implementing machine based instrumentation (MBI). These will be discussed in detail below.

Utility packages include word processing, grade book routines, bookkeeping and inventory control software. Developement tools would include special graphics authoring packages and assemblers. Programming languages such as BASIC, LOGO, Pascal and Fortran are well known words in the vocabulary of even the uninitiated computer buff. Time will not be taken here to discuss these topics which lie beyond our focus on computer applications in science education.

Machine Based Instrumentation

Until recently those who used computers as laboratory instruments were specialists who had considerable understanding of both the computer hardware and the software. Now a number of companies are offering a variety of options for the prospective user. These products might utilize the game paddle port of the computer as the interface, they might utilize a relatively inexpensive printed circuit card for interface or they modify the computer into a dedicated instrument.

Game Paddle Port Instrumentation

Most computers come equipped today with a very simple means for interfacing instrumentation. The game paddle port on the APPLE computer is a good example. On this one plug we find several inputs that can tell if something is switched on or off, several outputs that allow the computer to do the switching and several inputs that permit the computer to measure the electrical resistance of an external device. The variety of measurements that are possible in the laboratory with only this one interface capability is mind boggling. A simple example or two should help you to catch the vision.

A common measurement that is done in the physics lab is the measurement of the acceleration due to gravity. There are numerous techniques that are commonly used. But the obvious method of simply dropping the object and measuring the time of fall is rarely used. The time intervals are too short for accurate measurement with a stop watch. Computer to the rescue via the game paddle port. Several companies now sell a product that mechanically holds a ball in position for release when the computer sends the proper signal. Here is where we make use of one of the digital (on/off) outputs. Next we need to measure the time of fall. Many of you already know that any computer operates with a crystal controlled clock. It is therefore capable of very precise timing. Witness the digital watch that most of you are probably wearing. When the ball is dropped under the control of the computer, the measurement of time begins. When the ball strikes the target at the bottom of its fall a switch is thrown which tells the computer to stop timing. This is where we use one of the digital (on/off) inputs of the game paddle port. Of course digital information can be created by electronic as well as mechanical means. Photogates are often used to create electronic on/off information. They can be used by the computer for measuring the elapsed time for an object passing through the light beam. This information can be used to determine the speed of the object.

A thermistor is a device whose electrical resistance is related to the temperature. When a thermistor of the proper size is connected to the resistance measuring inputs of the game paddle portng upon them. Instant lightmeter! After collecting the data the graphics capabilities of the computer come in handy for displaying a graph of temperature versus time or light intensity versus time. A product is available from Technical Education Research Centers, 8 Eliot Street, Cambridge, MA 02138, which permits measurment of all of these variables and more. Can you see how one might measure the shutter speed of a camera or the response time of a student? For more of a challenge try figuring a way to measure and display the pulse rate of the user.

Interface Cards

Add-on interface cards begin by giving you more of what you already have with the game paddle port. Usually they will have digital (on/off) terminals that can be defined as either input or output by the software. They may also contain special timing capabilities and integrated circuits that perform analog/digital conversion. In the real world most experimental quantities vary continuously. This means they will produce electrical signals that vary continuously which are known as analog signals. The computer, on the other hand, is basically an on or off device. These two states are thought of, for purposes of doing computations, as either a zero or a one. The number system that is used is called binary because it has just two numbers in it, zero and one. At the fundamental level a digital computer is not equipped to deal directly with analog quantities. With the aid of analog/digital conversion devices it is possible for the computer to collect arbitrarily good digital approximations to analog quantities and store them away in memory or create a graphical display. The main advantage these add-on boards have over the game paddle port interface is speed. The conversion time for a typical low-cost analog to digital conversion chip is 15 microseconds. The analog input of the game paddle port takes 3 milliseconds to convert and then it will only look at quantities which can be represented by an electrical resistance. Speed of conversion is important whenever the computer is being asked to look at rapidly changing analog inputs such as sound waves.

For those interested in further study on the techniques used for this kind of machine based instrumentation Marvin L. De Jong has written a book called APPLE II Assembly Language which is published by Howard W. Sams & Co., Inc. Another book that treats this topic and the whole question of using computers in science education is Selecting and Using Microcomputers in Science Education by Jean L. Graef. This book is published by Cambridge Development Laboratory, 100 Fifth Avenue, Waltham MA 02154.

The Computer as a Dedicated Instrument

They are expensive but they are nice. Most schools can not afford to think about using computers in this way but it sure is fun to dream. Northwest Instrument Systems sells a plug-in device for the APPLE which converts it into a 50 Megahertz, dual channel, storage oscilloscope. This accessory costs an even kilobuck. But to buy comparable performance in a stand alone instrument would cost at least two or three times as much. This example is not unique. More and more commercial products are appearing that take advantage of the favorable cost/performance ratio of the microcomputer. The trend is sure to continue. It is time that our students began seeing the impact of high technology in their science classrooms. The computer is more than a device that is useful for teaching students how to program. It is actually an important laboratory instrument in the real world of science and engineering. Our students need to begin using it that way in their education.

Getting Started

Hopefully you are now eagerly chomping at the bit to get started using the computer in one or more of the ways we have discussed. Certainly the information presented here is inadequate to facilitate your new found zeal. So what can you do?

Self Study

In addition to those references already cited you should beg borrow or steal the March 1983 issue of Classroom Computer News. This magazine, published by Intentional Educations, Inc., 341 Mt. Auburn St., Watertown, MA 02172, contains several important articles on computers in the science curriculum. Another publication that should be on your must read list is Hands On published more or less quarterly by Technical Education Research Centers.

Most of you probably already subscribe to professional magazines which may contain a column devoted to computer applications, e.g. The Physics Teacher. Once you read these resources and look at the advertisements you will become acquainted with some of the companies that supply software/hardware products for science education. Their catalogs often contain very helpful information on the functions performed by their products. But suppose you need a little more assistance. The magazines and books, though the words appear to be English, contain sentences that sound like Greek. You need a translation. This is where a workshop is useful.

Workshops and Courses

Both of the organizations with which I am affiliated have offerings to meet this need. In addition, these services are offered by Cambridge Development Lab and Technical Education Research Centers. The American Association of Physics Teachers has several hands on workshops that can be shipped to your location where they can be presented under the supervision of a qualified leader from your locality. Most of the people who are already in the field are quite willing to help the beginner. Ask questions. That is the best way to learn.

7

THE USE OF COMPUTERIZED DATABASES TO LOCATE INFORMATION ON SOFTWARE

Laurene E. Zaporozhetz

Introduction

Computerized literature searching gives educators alternatives for locating information. The number of databases that can be searched by computer grow daily. Databases for software are recent additions to this field. Four to be discussed at this presentation are: (1) Resources in Computer Education, (2) DISC, (3) International Software Database, and (4) Microcomputer Index.

Resources in Computer Education

This database covers information from 1979 to date concerning the state of the art in school computer applications. It includes data on software producers and packages, with bimonthly updates of 200-300 records. It can be searched a number of ways including descriptors, (using the Thesaurus of ERIC Descriptors) grade level, mode of instruction, hardware type, and system requirements. The complete record includes a standard citation, all of the above information, plus an evaluation, evaluation summary, instruction objectives and a contact information address. A print counterpart does not exist. Resources in Computer Education can be searched through the Bibliographic Retrieval System.

DISC

DISC covers information from January 1982 to date with cover-to-cover indexing of journals in microcomputing with peripheral coverage of minis. It has bimonthly updates of about 300 records. It can be searched by author, title, and source; as well as table of contents, or publisher name. The complete record includes a complete citation, or the complete table of contents from a microcomputing journal. A print counterpart does not exist. DISC can be searched through the Bibliographic Retrieval System.

International Software Database

The International Software Database covers information from 1973 to the present in the following major areas of software: commercial, educational, industrial, personal, professional, scientific, specific industries and systems. Additional records are added on a monthly basis. It can be searched by machine, operating system, application, vendor and price. The complete record includes a full description, date of release, warranty indication, compatible systems, and price. References to independent reviews are included when available. A print counterpart does not exist. The International Software Database can be searched through the Dialog Information Retrieval Service.

Microcomputer Index

The Microcomputer Index covers information from January 1981 to the present containing citations to the literature on the use of microcomputers in business, education, and the home. Magazine articles as well as software and hardware reviews, new product announcements, and book reviews are included. It has monthly updates of approximately 700 records. It can be searched by program listing, geographic location, journal name, and descriptor. The complete records include a brief abstract. The database corresponds to the quarterly printed publication Microcomputer Information Services. The Microcomputer Index can be searched through the Dialog Information Retrieval Service.

Preparing for a Search

To begin a search, first clarify your question and the type of search you want. For example, do you want a broad search for a lot of material, or a quick search for a few articles? Second, choose relevant databases. Third, divide your question into parts or word groupings and develop a list of keywords to describe your information needs. If applicable, to the database, list controlled terms from the Thesaurus for each concept.

The final step is to use Boolean Logic to formulate logical groups for the parts of your search. The term "OR" links your synonyms and broadens your search. The term "AND" overlaps concepts that appear within the same citation. The term "NOT" excludes a group of terms or bibliographic limitations.

For further information

The appended bibliography identifies the major bibliographic search systems, books and articles explaining the search process, and database directories.

Selected Bibliography

Major Online Bibliographic Information Retrieval Systems

Bibliographic Retrieval Services (BRS)
702 Corporation Park
Scotia, NY 12302
Phone (800) 833-4707

Dow Jones News/Retrieval
22 Cortlandt Street
New York, NY 10007
Phone (212) 285-5225

Lockheed Information Systems (DIALOG)
3460 Hillview Ave.
Palo Alto, CA 94304
Phone (800) 227-1960

National Library of Medicine (MEDLINE)
8600 Rockville Pike
Bethesda, MD 20014
Phone (800) 638-8480

New York Times INFORMATION BANK
Mt. Pleasant Office Park
1719-A Route 10
Parsippany, NJ 07054
Phone (800) 631-8056

SDC Search Service (ORBIT)
2500 Colorado Ave.
Santa Monica, CA 90406
Phone (800) 421-7229

Journals

Database: *The Magazine of Database Reference and Review.*
Online Incorporated
11 Tannery Lane
Weston, CT 06883

Online: *The Magazine of Online Informations Systems.*
Online Incorporated
11 Tannery Lane
Weston, CT 06883

Online Review
Learned Information Incorporated
The Anderson House STOKES ROAD
Medford, N. J. 08055

Books and Journal Articles

Blue, Richard I. "Questions for Selection of Information Retrieval Systems." *Online Review*, 3, 1 (March 1979), pp. 77-84.

Bourne, Charles P. "Online Systems: History, Technology, and Economics." *Journal of ASIS*, 31, 3 (May 1980), pp. 155-160.

Fenichel, Carol H. and Hogan, Thomas H., *Online Searching: A Primer*. Medford, N. J.: Learned Information, 1981.

Jewell, Sharon and Brandhorst, W. T. "Search Strategy Tutorial: Searcher's Kit." Washington: National Institute of Education, October 1973, ERIC Document number ED-082-763.

Rouse, Sandra H. and Lannom, Lawrence W. "Some Differences Between Three Online Systems: Impact on Search Results." *Online Review*, 1, 2 (June 1977), pp. 117-132.

Smith, Sally M. "Venn Diagraming for Online Searching." *Special Libraries*, 67, 11 (November 1976), pp. 510-517.

Directories

Hall, J. L., Online Information Retrieval Sourcebook. 2nd edition. London: ASLIB, 1981.

Hoover, Ryan E., et al., The Library and Information Manager's Guide to Online Services. White Plains, New York: Knowledge Industry Publications Incorporated, 1980.

Kruzas, Anthony T. and Schmittroth Jr., John, editors, Encyclopedia of Information Systems and Services. 4th edition. Detroit: Gale Research Company, 1981.

Wanger, Judith, Landau, Ruth N., and Berger, Mary C., Directory of Online Databases. Santa Monica: Cuadra Associates, 1981. Semi-annual updates.

Williams, Martha E. and Rouse, Sandra H., Computer-readable Bibliographic Databases: A Directory and Sourcebook. Washington: American Society for Information Science, 1982.

Bibliographies

Hawkins, Donald T., "Online Information Retrieval Bibliography, 1965-1976." Online Review, 1, Supplement (January 1977), pp. S11-S122.

______., "First Update." Online Review, 2, 1 (January 1978), pp. 63-106.

______., "Second Update." Online Review, 3, 1 (January 1979), pp. 37-73.

______., "Third Update." Online Review, 4, 1 (January 1980), pp. 61-100.

______., "Fourth Update." Online Review, 5, 2 (April 1981), pp. 139-182.

______., "Fifth Update." Online Review, 6, 2 (April 1982), pp. 147-208.

Specific Database Information

Resources in Computer Education

Producer: Northwest Regional Educational Laboratory
300 S. W. Sixth Avenue
Portland, OR 97204
Contact: Barbara Rozell
(503) 248-6800 Ext. 551

DISC

Producer: BRS
1200 Route 7
Latham, N. Y. 12110
Contact: Customer Service: (800) 833-4707
(518) 783-1161 (in New York)

International Software Database

Producer: Imprint Software Ltd.
1520 South College Avenue
Fort Collins, CO 80524
Contact: Aneta Diekroger
(800) 525-4955
(303) 482-5000 (in Colorado)

Microcomputer Index

Producer: Microcomputer Information Services
2464 El Camino Real, Suite 247
Santa Clara, CA 95051
Contact: Joe Ward
(408) 984-1097

MICROCOMPUTER APPLICATIONS OF CAREER INFORMATION

Bruce McKinlay

Computers have become standard tools in business, manufacturing and commerce. In educational institutions, administration has made far more and better use of this information processing machine than has instruction. A review of the use that has been made of computers in schools quickly reveals career information to be among the first and largest direct use of computers by students. A decade of delivering career information to students by computer has taught several lessons about schools, students, information, and technology that can be valuable to educators who hope to shape or benefit from the new microcomputer era in education.

This paper describes briefly the use of career information systems in schools and provides some check points for the selection of microcomputer equipment and software for career information.[1]

Computer-based Systems for Career Information

Whatever practical task it may be assigned, the computer is at its roots an information processing machine. Thus it is not surprising to see the computer put to use storing, sorting, and displaying career information for students. Occupational, labor market, and educational information has traditionally been accessible only to the economists and planners who know their way around federal and state statistical agencies. But the organizations that operate career information systems have changed that, and the computers--until now the big macro and mini time-shared computers--have been some of their principal tools.

The idea of letting individuals use computers to sort through masses of occupational and educational information began taking form in the early 1970's when such programs moved beyond the pilot project phase. Last year a large-scale, nationwide survey revealed that, of the four sources of career information valued most highly by secondary school staff, two were computerized information systems. Along with the venerable <u>Occupational Outlook Handbook</u> and <u>Dictionary of Occupational Titles</u> published by the U.S. Department of Labor, were the Guidance Information System (GIS) and the Career Information

[1] "Career information systems" are intended to provide informational support to individuals in two realms of career development, making educational plans and deciding about occupations to explore.

System (CIS).[2] Beginning in Oregon just twelve years ago, state and local system operators now compile occupational and educational information and deliver it to ten thousand schools and social agencies. Approximately 4 1/2 million students, teachers, counselors, and clients used the systems this year.[3]

All of the successful systems for career information have several features in common:

- information that has been analyzed and written for the average person, not statistical data;
- both occupational and educational information presented in common sense categories such as "pilot and flight engineers "computer science," and "apprenticeship";
- the ability to sort through the occupations and colleges and generate various lists, as well as the ability to display tl information about a particular topic;
- information that is continuously updated;
- printed copy for a person's further study after leaving the computer terminal.

Within the field, a distinction is made between computerized "information systems" and computerized "guidance systems." Information systems are those that concentrate on delivering accurate information and making it easily accessible for a variety of instructional and counseling applications. Guidance systems, on the other hand, computerize either a particular guidance process or selected guidance activities. The widely-used Career Information System and Guidance Information System are both on the information side of this spectrum.

Organizationally, there has been a major effort by both federal and state governments to establish an organization in each state to perform two vital functions. First to analyze local as well as national information, and secondly, to provide training and follow up to schools and other institutions on the installation and effective use of their systems. In addition to using computers to deliver the information to local schools and agencies, a majority of these career information system operators also use manual delivery means such as needle sort cards and books to make the system available in small and rural places. Currently there are state-based systems in 36 states, according to the Association of Computer-based Systems for Career Information, the professional association in the field.

[2] Warren Chapman and Katz, Martin R. <u>Survey of Career Information Systems in Secondary Schools: Final Report of Study 1</u> (Princeton: Educational Testing Service, 1981).

[3] Association of Computer-based Systems for Career Information, <u>Level of Service Report 1971 - 1983</u> (Eugene: Association of Computer-based Systems for Career Information, 1983).

Lessons About the Use of Computers in Education

Experience with the major career information systems in schools throughout the country leads to a number of observations about the educational use of computers.

1. Students like using computers. In the post-Pacman era, this observation seems trite. Today, few people would be surprised that 92 to 95 percent of both general population and disadvantaged high school students rated a career information system easy to read, easy to use, and fun to use.[4] Most students are able to operate the computer terminal and the career information software unaided. They not only can, they do; as many as two-thirds of a high school student body will use a system during a school year.

The important point is that, even with educational computer games as alternatives on the school's computer, students choose to use educationally substantial programs such as computerized career information systems.

2. Professional leadership is important to full use of information systems. The better career information systems are attractive and easy to use, and students are naturally curious about occupational and educational opportunities. Both GIS and CIS provide printed user material to assist system users. Unaided, many students will use and benefit from the system. Even so, the educational effectiveness of the systems is greater if they are complemented by other materials and their use is integrated into guidance and instructional activities. Many schools and colleges have career information centers providing printed material and career development activities along with the basic computerized system.

Like any powerful tool, a computer program has to be used to be effective. State-level encouragement and assistance are needed to help institutions install, promote, and use new technologically-based tools such as computerized career information systems. In some schools numerous classes, groups, and other activities use the system effectively; in others only aggressive students find this resource. Local site leadership makes the difference.

3. Computerized information systems are cost effective. On a cost-per-user basis, computerized career information is a low cost option. Books alone tend to stay on library shelves, while counselor time is too scarce and too costly to be used simply for retrieving and delivering factual information. In practice, a career information center adds a low cost information service to the limited personal and educational counseling that is possible in a student service office, greatly increasing the number of

[4] Bruce McKinlay and Michael R. McKeever, "The Career Information System: A Decade of Developmental Research," Arrive: Annual Review of Research in Vocational Education, Volume One, ed. Tim L. Wentling (Urbana, Illinois: University of Illinois and Illinois State Board of Education, 1980).

students who can be served by the department. The cost of equipment and space must always be compared with the costs of doing nothing or having professional staff provide the service. In the case of career information systems, the accounting favors the computerized approach and helps make the entire student service program more cost effective.

4. <u>Career information systems provide an information base for a variety of instructional as well as guidance functions.</u> Initial career planning is one of the important tasks of adolescence and young adulthood, just as career change is becoming an option for older adults. But career guidance is only one of the missions of educational institutions whose graduates need to function as learners, citizens, consumers and individuals as well as producers. One of the lessons of career information systems is that both the information base and the computerized delivery medium can contribute to instruction as well as to guidance. Many schools and colleges have made creative use of this resource in mathematics, computer science, business, social sciences and other instructional areas.

The point becomes even clearer if we turn for a moment from the specific case of career information system. Take the Educational Resources Information Clearinghouse (ERIC). This is one other mainframe-based information system that sorts, lists, and displays information about topics (in this case research and curricular materials) that are important to education. As we think about not only career information systems but information systems generally, we see more clearly the support they can provide in the school.

Computer resources, especially computerized information, enrich a wide range of instructional activities besides the one major function that is originally intended. Tools of this kind are best thought of as resources of the institution, not just of single departments.

Equipment Requirements

Schools and colleges planning their use of computers can anticipate three functions for computers in education: administration, instruction, and information systems. In administration, reliability and security are prime requisites of a satisfactory system. Vital personal and financial records, often massive in size, must be kept invulnerable to loss or damage. Access to them must be restricted to people with legitimate need to know. These reliability, space, and confidentiality needs have been met largely through the use of centralized mainframe computers.

In instruction, microcomputers of all types are being put to use. We may not yet have seen the ideal instructional computer and one wonders whether it will have some of the features that have made the textbook so ubiquitous - portability, durability, and cost low enough to permit each student to have one. The relatively small memory requirements of individual instructional programs and the value of individual copies of instructional materials suggests such a device.

The information system has somewhat different requirements as well. Information is bulky; a state career information system's files often contain five to seven million characters of information. So, like administrative uses, information systems need large amounts of storage.

On the other hand, access to information files should not be restricted--the files are intended for the widest possible use. In this respect the information machine should be more like the teaching machine, easy to use and readily accesible.

As with other useful tools, there are classes of computers, making some better suited for one purpose than another. David Moursund summarizes the distinctions well. In his School Administrator's Introduction to Instructional Use of Computers, he points out:

> Computer hardware is general-purpose in the sense that it can be used on a wide variety of problems. But some hardware systems are specifically designed to meet business application needs while other systems are specifically designed to meet scientific research needs. An inexpensive microcomputer may be an excellent aid to instruction and to student learning but be nearly useless as an educational administrative aid. Conversely, a computer designed to fit the needs of school administrators may be quite inadequate in meeting the needs of students.[5]

On the questions of storage capacity, even the smallest of the mainframe computers (usually referred to as minicomputers) with remote terminals support well an information system's large files. However, their telephone connections to the terminal and complex access codes are not easy for inexperienced users to master, even though the information system program itself usually is, and the cost of telephone access is prohibitive in many rural locations. Thus, while career information systems running on mainframe computers have become widely used in education, access has been an annoyance and the full use of computerized information systems has been limited by the telephone hook-up costs and sometimes computer network costs.

Microcomputers solve the access problem and can greatly reduce cost, though not every microcomputer can manage the large information files. A career information system program can be brought up on a microcomputer with a few easy steps and no access codes. Regarding file space, a career information system requires microcomputer mass storage greater than a floppy disk can hold, however. It would take 30 or more of the common 5 1/4 inch floppy diskettes to hold the files of a typical state career information system. Thus, most microcomputer-based career information systems will store that information on hard Winchester-type disks that have a capacity of ten million characters or greater.

[5] David Moursund, School Administrator's Introduction to Instructional Use of Computers (Eugene: International Council For Computers in Education), March, 1982.

The cost advantages of microcompters for career information delivery can be impressive. The high school in Firth, Idaho, north of Pocatello serves 300 students with the Idaho Career Information System on a microcomputer. To serve the same students using the nearest time-shared computer would have cost three times as much, mostly in the form of a two thousand dollar per year telephone bill.

The saving is not limited to small, rural locations. Colorado Springs School District was able to replace time-shared delivery to their five high schools and three junior highs with a microcomputer the Colorado Career Information System and save $35,000 in telephone and time-shared computer costs. Such savings can quickly cover the costs of microcomputer equipment.

The independence of the microcomputer from the telephone has logistical as well as financial advantages. Lane Community College in Eugene, Oregon, serves adult learners in neighborhoods and small communities via a mobile van, which now contains a microcomputer to deliver part of the Oregon Career Information System.

Hardware Requirements for a System of Career Information

The microcomputer industry is diversifying. Low-power computers such as the Texas Instrument 99/4A, Timex Sinclair 1000, and the Commodore Vic-20 are selling for barely a hundred dollars, while IBM and other firms are producing very capable models for business use at three to ten thousand dollars. Apple II and TRS-80 remain in the middle area, but the industry may be segmenting into a low-cost, low-power "home" segment and a higher priced and more powerful "business" segment. A career information system needs a mid-range or business level machine to run an entire system. Typical hardware requirements for career information systems are: 64K main memory, CP/M operating system, one floppy disk drive for loading updated information, 10 megabyte hard disk for storing information files, and a printer so the system user can have a copy of the requested information.

Floppy diskettes are more common in schools than hard disks. However floppy diskette drives account for only half the total mass storage sales, while Winchester and other hard-disk storage account for the other half. The market penetration of Winchester-type disk drives is expected to triple by 1990.[6] There are useful career education activities that will fit on floppy diskettes, but there is no way to squeeze a full set of occupational and educational information onto one or two floppy diskettes. Thus, it is worthwhile to remember that anything on a floppy diskette is just part of a system. An example is "Micro-QUEST," an occupational sorting activity from the Career Information System. Micro-QUEST runs on a single diskette, but it has to refer users to the books of the Career Information System files that come with it in order to make a full system. There are several such partial products, on one or two floppy

[6]"Disk Market Overview," <u>Mini-Micro Systems</u>, XVI, 5, (Spring 1983), p. 71.

diskettes but none contains full information files, which are, afterall, the primary reason for installing a career information system.

Most educational computing planners recommend that you first decide on the educational function to be supported, then find the most appropriate software, and finally buy equipment to support the educational purposes and software reqauirements. Adam Osborne, in a clear and intelligent book called An Introduction to Microcomputers, Volume Ø, The Beginner's Book, points out, "A microcomputer without software is similar to a record player without records. Both are virtually worthless." The point is a good one; a usable system consists of both hardware and software. The analogy cannot be carried too far, however. Any good record player will play just about any record you can buy, and most even allow you to adjust for records of different sizes and speeds. The same is not true for computers. No microcomputer will run every kind of software, or even very many kinds. So in the microcomputer world it is important to know what kind of "music" you want to play before buying a machine to play it.

As we have seen, administration and instruction have quite different hardware requirements, while information systems have some requirements in common with both. That is part of the reason career information systems have done well sharing mainframe computers with administrative activities. As we look into microcomputers, we see that instruction and information systems need many of the same capabilities, the primary difference being the information system's need for more mass storage than instruction requires. The complementary purposes and hardware requirements of instruction and information systems recommend joint use of equipment whenever that is cost effective.

Features of Good Information Systems

What, then, should one look for in choosing a career information system? George Morrow is the founder and president of Morrow Designs, the manufacturers of high quality business computers. He would be one of the first computer engineers to agree on the importance of good software selection. "I firmly believe that software is more important then hardware," Morrow is quoted as saying.[7]

In an information system the computer is a delivery medium, a vehicle for moving the information from research office to user site. For an information system to be an effective system three elements must be adequately provided for - content, delivery, and use. An attractive computer program based on outdated or inaccurate information is worse than no program at all because it misleads students. And a system that goes unused or is used inappropriately is not worth its cost.

A consumer can judge an information system in the same way one does any other product, by examining its contents, seeing how well it works, and checking the organization that produced it. Some guidelines for evaluating career information software are identified here.

[7] Computer Retail News, May 2, 1983.

Content: The Information Itself

Comprehensive The information should cover all types of occupations and educational opportunities. Particularly look for specific, local information. Is information on specific schools, occupations, and local labor markets included? Are vocational schools and post graduate schools included as well as four-year colleges? Are hiring requirements, wages, and local outlook reported along with job duties?

Current The information needs to be applicable at the time of use. Is new information entered as soon as it becomes available? Is all information reviewed at least once a year? How many professional information analysts work year-round maintaining the information? Are updates distributed to the field more than once a year?

Impartial The information should be factual, not reflecting the biases of a particular writer or organization. Such personal judgments should be left to the system user. Is racist, sexist, and age-biased language scrupulously avoided? Does the system avoid status distinctions between one type of occupation and another or one kind of education and another?

Delivery: Hardware and Software

Both sorting and information display A career information system will allow the user to sort through the files to create lists of occupations or schools. It will also provide comprehensive, current, and impartial information about the options listed. Does the system provide both sorting and full information display? Are the sorting components based on empirical data, not just someone's opinion?

Easy to use To be cost effective and to have greatest impact, the information system should be operable by students and clients, allowing teachers and counselors to spend thier valuable time interpreting information and helping students plan. Can your students or clients follow the directions and operate the system?

Print copy People need copies of information for further study. Does the system provide printed user material? Does it have a printer for copies of computer displays?

Use: Support for User Sites

Training and technical assistance Someone familiar with the system should provide staff training and help with integration of the system into teaching and counseling functions. Are training and technical assistance regularly available? Are there annual workshops in the area to update counselors and train new staff? Where do you get help when you need it?

Local support organization Local information and local training are essential ingredients of a career information system. They are most effective when provided within the state. Is informaton analysis conducted in the state? Are system staff always available in the state to answer questions and provide training? Is management of the system located within the state?

There are many specific points on which a career information system could be judged. The few features listed above are some of th most important and some of those that separate complete systems from partial ones.

Specific standards for computer-based career information systems are published by the Association of Computer-based Systems for Career Information in their Handbook of Standards for Computer-based Career Information Systems[8]. A directory of career information systems and a publications list are also available from the ACSCI Clearinghouse.

Summary: The Application of Microcomputers to Career Information

Along with administration and teaching, information constitutes one of the three major applications of computers in education. Among information systems available on mainframe systems, career information has become one of the most prominent in schools and colleges.

Microcomputers are increasing the number of places where computerized sorting and display of career information is feasible by reducing information delivery costs. Thus planning for microcomputer use should include information system as well as instruction and administration.

A career information system is a much larger piece of software than an instructional package, mostly because of the space required to store accurate and usable information files. Information systems need equipment that is more akin to a business microcompuer than a home game computer. While parts of systems may be run on a microcomputer with a floppy disk drive, such a product is only part of a system. Full systems require mass storage of the type provided by Winchester hard disks.

With the exception of this larger space requirement, information systems and computer-aided instruction have much in common. They serve kindred purposes and have many other hardware requirements in common. A well-conceived plan for computers in education should anticipate the multiple use of career information in instruction as well as guidance. It could also prove cost effective for instruction and the career information system to share hardware.

[8] Association of Computer-based Systems for Career Information. Handbook of Standards for Computer-based Career Information Systems. (Eugene: Association of Computer-based Systems for Career Information), June, 1982.

For further information about computerized career information systems, write:

ACSCI Clearinghouse
Hendricks Hall
University of Oregon
Eugene, Oregon 97403

A MODEL FOR DEVELOPING AN ELEMENTARY SCHOOL COMPUTER SCIENCE CURRICULUM

JOHN C. ARCH
May 27, 1983

DEFINING THE PROBLEM

There can be little doubt that more funding for computer science education is coming to our schools.

Recently, the governor of Tennessee, Lamar Alexander, called for increased expenditures on computer science education in that state, with the eventual goal of placing at least one microcomputer in every school in the state [The Tennessean]. This in a state not known for leadership in the area of technological education.

The federal government is also making increased financial commitments to upgrading technological education. For example, the Emergency Mathematics and Science Education bill, recently passed by the House, would give $250 million to schools to improve science and math curricula [NY Times].

In 1982 the Houston Independent School District created the nation's first Department of Technology. With a budget of over $4 million and a staff of forty this department has the dual tasks of overseeing the introduction of computer science education to Houston's 190,000 students and training to the districts's 14,000 teachers [Sturdivant, 1983].

The problem for computer science educators is to determine the best method for spending their share of this increased funding.

To date the thrust of instructional computing has been threefold, as perhaps best formalized by Robert Taylor's [1980] book TOOL, TUTOR, TUTEE. For the most part students have been taught to program (the computer as tutee), they have been taught to use the computer as a powerful problem solving tool (the computer as tool), and they have been taught by the computer, through some form of computer assisted instruction (the computer as tutor).

The common thread running through these three strands is the computer itself. Each requires students to sit in front of a terminal or micro. Aside from the fact that it will be extremely expensive to buy the hardware and software needed to fully develop these uses of the computer, each of these approaches misses what may be the key element in the development of a computer literate populace, namely the ability to think like a computer scientist. In our rush to use the computer for instructional purposes we may have overlooked this important aspect of computer science education.

PROBLEM SOLVING AND COMPUTER SCIENCE

One possible reason computer science has developed so rapidly into a separate discipline is that it requires it own unique set of problem solving skills. Indeed, it has been claimed that one purpose of professional education is to train problem solvers in particular fields [Cyert,1980]. Computer science can no longer adequately be taught simply as an adjunct to some other discipline, such as mathematics. It must be recognized that computer scientists solve problems in different ways than do mathematicians, writers, lawyers, chemists, and so on. It is a belief expressed repeatedly throughout this paper that these problem solving skills are a key element of what is essential about computer science. Furthermore, it is contended here that these problem solving skills can be defined, isolated, and distilled into a curriculum that elementary school students can understand and that this curriculum can be taught without the purchase of a great deal of expensive hardware and software. This paper will outline a procedure for developing such a curriculum.

THE IDEAL

The ideal situation for delivering computer science education to elementary students would be to have a microcomputer lab and the trained personnel to operate and maintain it in each school. The problem with reaching this ideal is twofold - money and time. A great deal of money would be needed to buy the requisite hardware and a great deal of time and money would be needed to train the needed personnel. Eventually, this ideal will be attained but not in the immediate future. It is clear that an interim solution is needed, one that can expose young children to the problem solving skills required by computer science without requiring schools to purchase too much hardware too soon.

WHAT HAS BEEN DONE

CURRICULUM DEVELOPMENT

Much of today's writing on curriculum development can be traced back to Ralph Tyler's book, BASIC PRINCIPLES OF CURRICULUM AND INSTRUCTION, first published in 1949. In it Tyler enumerated his now famous rationale for curriculum development:

> Four fundamental questions...must be answered in developing any curriculum and plan of instruction. These are: (1) What educational purposes should the school seek to attain? (2) What educational experiences can be provided that are likely to attain these

purposes? (3) How can these educational experiences be effectively organized? (4) How can we determine whether these purposes are being attained? [Tyler, 1949]

Curriculum planning begins with needs assessment and passes through a series of steps before teachers begin implementation. A widely accepted model for developing curriculum was proposed by Taba in 1962. She proposed the following seven steps, based closely on the work of Tyler:

Step 1: Diagnosis of needs
Step 2: Formulation of objectives
Step 3: Selection of content
Step 4: Organization of content
Step 5: Selection of learning experiences
Step 6: Organization of learning experiences
Step 7: Determination of what to evaluate and of the ways and means of doing it.

It has been argued that these and similar models of curriculum development are based too mucch upon the authors' own opinions rather than upon the findings of rigorous empiracle research [Goodlad, 1980]. Indeed, Tyler's book contains not one cite, nor even a bibliography. But as Goodlad points out there has not been a great deal of research into curriculum development. Hence, the plethora of opinion in much of the writing on the subject.

An extreme view is that of Papert [1980], who argues that most of today's ccurriculum planning is misguided. Papert, expanding upon the theories of Piaget, calls for what he terms a theory of learning without curriculum. To Papert this new learning means "supporting children as they build their own intellectual structures with materials drawn from the surrounding culture." [Papert, p 32]. He sees the microcomputer in general and LOGO in particular as having the potential to provide a great deal of this needed support.

Despite the shortage of research into curriculum planning and development it is obvious that many millions of children will arrive at the schoolhouse door each schoolday morning for as far into the foreseeable as one cares to look. There a curriculum of some sort will await them. The process we call education will not suspend itself while educators develop a universally accepted definition and model of curriculum [Goodlad, 1980].

Curriculum experts have long debated the role that should be played by scholars in the development of curriculum. At the higher levels of American education, scholars have exerted almost complete control over curriculum. For instance, academic specialists have totally controlled most university curricula. Indirectly, they control much of secondary curriculum, since it is universities that set the standards for the advanced placement and achievement tests [Pratt,1980].

King and Browell [1966] took the extreme view that only scholars have the training and knowledge to determine what subjects should be taught, not just in universities but in schools at all levels. Others take an opposing tack:

> Scholars, as such, are incompetent to translate scholarly material into curriculum. [Schwab, p.1]

To say the least, the state of the art regarding the input of scholars into curriculum development is unsettled. No one, however, has gone so far as to say that scholars should have no input. The consensus seems to be that some input is needed from scholars, but that other legitimate sources of input exist and should be sought out. These other sources of input include:

1) skilled and experienced teachers, curriculum design experts [Pratt, 1980]

2) administrators, principals, classroom teachers, supervisors, lay citizens [Shuster and Plaghofl, 1977]

3) teachers, curriculum leaders, students, parents, consultants [Wiles and Bondi, 1979]

4) teachers, students, parents, principals, central office personnel, outside resource experts [Morley, 1973]

5) students, the milieus in which learning occurs, teachers, curriculum experts [Schwab, 1973]

The model for curriculum development developed in this paper will concentrate upon professional computer scientists, classroom teachers, and students.

CURRICULUM DEVELOPMENT IN COMPUTER SCIENCE

If anyone has developed a unit which attempts to do what I propose to do in this paper, it has not been reported in the literature. Many authorities recognize that something must be done to bring computer science education into the elementary schools, but attempts to do so invariably have different goals than what is proposed here.

An important project in the area of elementary school computer science education is AN APPROACH TO INTEGRATING COMPUTER

LITERACY INTO THE K-8 GRADES [Hunter, 1980]. This project is is being funded by the National Science Foundation. Begun in 1980, with a termination date of September 1983, this project seeks to find, develop, and disseminate information about computers into the nation's elementary schools. The project seeks to:

> '...infuse computer-related skills into the traditional curricula of elementary and junior high school science, social studies, and mathematics courses.' [Hunter, p. 3]

While the goals of this project are important, they ignore the development of skills that are essential to developing an understanding of the thought processes and problem solving skills essential to computer science.

A project developed in 1974 [Robinson, 1974] for the Oklahoma State Department of Vocational and Technical Education had, among other things, the following objectives for students:

- to identify the basic parts of a computer;
- to apply rules for raising base ten numbers to a power;
- to change base ten numbers into base two numbers;
- to read a computer card; and
- to write a flow chart.

These may or may not be worthy goals, but they certainly do not impart any insight into the thought processes that comprise the foundation of computer science.

A project developed at Stanford University in 1975 got closer to some important issues of computer science [Weyer, 1974]. The goals of this project were to teach programming to students between ten and fifteen years old, and by so doing to impart certain key concepts about computer science. Among the concepts the developers sought to impart were the following:

- literals;
- names and values;
- evaluation and substitution;
- stored programs;
- decisions;
- procedures;
- argument passing;
- functions;
- recursion; and
- iteration.

There are at least two problems with this approach: (1) it requires a staff that is proficient in programming, and

(2) it requires too much hardware and software to be practical for all but the wealthiest school districts. Given the shortage of computer science educators and the current state of the economy, these are serious shortcomings.

Another approach might be termed the computer literacy approach. Many curriculum guides mention that students should be taught 'computer literacy'. For instance, the PRISM (Priorities in School Mathematics) Project of the National Council for the Teaching of Mathematics, includes computer literacy as one of the nine strands it recommends schools stress during the 1980s [PRISM, 1981].

While 'computer literacy' is defined in many ways, it generally includes a brief introduction to computer facts (terminology, history, and so on), an overview of the impact of computers upon society, and a brief introduction to programming [Graham, 1983; Stern and Stern, 1983; Frates and Moldrup, 1983]. These are each worthy goals, but it is my belief that knowledge of the thought processes essential to computer science is also a worthy goal and a more attainable goal for the immediate future.

PROBLEM SOLVING

In the area of problem solving there has been much done but there is still a great deal of uncertainty and confusion. Greeno [Tuma,1980] believes that students need a knowledge base before they can solve a problem in a particular subject area, but he criticizes most attempts at teaching problem solving as being too narrowly focused on subject matter. I see this narrow focus as a fatal flaw in much of the computer curriculum that has been developed to date. Greeno calls for more effort to be expended in teaching general problem solving skills which transcend specific subject matter. The problem with this approach is deciding which techniques can be transferred from one type of problem to another. This paper presents a methodology for developing a curriculum that aims to teach some of the main problem solving techniques of computer science to elementary school children.

But can general problem solving skills be taught? The answer is that we do not know. Rubinstein has developed a course at UCLA which he claims will train college students in general problem solving skills [Rubinstein, 1980], but Reif criticizes Rubinstein's lack of empirical evidence to support his claims, despite the fact that the course has been taught for nearly a decade [Reif, 1980]. Larkin [1980] claims that research into general problem solving is hampered by the fact that problem solving cannot be measured using the traditional quantitative techniques of educational research. Instead, we need detailed protocol research into how students actually go about solving problems.

Without the knowledge that such research would provide we do not know enough about problem solving to teach general problem solving skills. The problem with teaching general problem solving skills is summed up by Hayes [1980, p 146]:

> If we are to do evaluation of a problem-solving course well, we have to show that the students spontaneously use the skills we have taught them outside the confines of the classroom in which they were learned.

Until we can show that this transfer of learning is possible we will not be able to claim that it is possible to teach general problem solving skills.

OTHER LITERATURE

Far too many educators seem bedazzled by microcomputer technology. Witness, for instance, the professional literature where computers are often heralded in terms akin to a fundamentalist preacher praising the second coming. Words and phrases such as "fantastic", "greatest human advance since the printing press", "education's next great revolution" abound. The entire October 1979 issue of EDUCATIONAL TECHNOLOGY typifies this reverence. Article after article sings the praises of the new technology, somehow assuming that microcomputers will improve education by their mere presence and we had better hurry, hurry, hurry to get micros and computer literacy into the schools before it is too late. No article deals with the thought processes that underly what it is computer scientists do. None deals with the problem solving skills that computer science can help build. It almost seems that some educators become so excited by computers that they do not want think before they act for fear of finding out that all is not what they want it to be.

WHAT NEEDS TO BE DONE

To construct a curriculum that would get to the heart of computer science problem solving skills the following model is proposed:

STEP 1: Define the Thought Processes Essential to Computer Science

To discover the thought processes essential to computer science a good place to begin is the faculty of a university computer science department. Each faculty member could be asked to define the thought processes most important to his or her specialty. The interviewer must keep in mind what s/he is after, namely the problem solving skills and mental disciplines that will be teachable to students in elementary school.

STEP 2: Develop Activities Suitable for Elementary School Students.

These activities should be based upon the processes defined in Step 1. They should be shown to a number of elementary school teachers, curriculum specialists, professors of education, and so forth for the purpose of soliciting their criticisms and comments.

Possible sources of activities include:

(a) Teachers currently in the classroom:
Classroom teachers are very resistent to curriculum imposed on them from above. To avoid such problems it is essential to involve classroom personnel in the curriculum process as early as possible. Early involvement also takes fullest advantage of the considerable expertise possessed by teachers. They are education experts and their opinions and ideas should be actively pursued in developing a unit such as that proposed here.

(b) Graduate students in computer science education:
If you have access to a university that trains graduate students in computer science education take full advantage of the situation. Most students in this area have had extensive classroom experience to which they have added advanced training in computer science education. Such people are a potential goldmine of useful information.

(c) The literature:
Many popular journals, such as The Computing Teacher, Electronic Learning, Personal Computing, Creative Computing, and so on, contain articles on instructional computing. Occasionally, these articles contain an activity which could be of some help in preparing this dissertation.

STEP 3: Field Test the Activities in Elementary School Classrooms.

Find one or more elementary school classrooms in which to try out the activities that have been developed. This will help eliminate as many 'bugs' as possible before the attempt is made to train teachers to use the activities.

The unit being developed by the author will last at least two or three weeks for approximately forty-five minutes to an hour each day. Each particpating teacher should be asked to critique each activity prosented to his or her class. In addition, each teacher should be interviewed to discover his/her opinions of the activities.

STEP 4: Train a Selected Group of Teachers to Use the Activities Developed and Refined Previously.

A one-week workshop should be more than enough to train a small group of teachers how to implement the activities previously developed and field tested. A possible prerequisite for this workshop could be some experience with computers, such as a beginning programming course or the equivalent, but even this requirement should be waived if a teacher shows enough initiative to sign up.

STEP 5: Develop Follow-up Activities.

If the goal of inservice training is to change teacher behavior, then a viable follow-up component must be added to the workshop [Keating, 1983]. It is beyond the scope of this paper to go into detail here. Those who are interested can see Keating [1983] for a detailed follow-up program designed specifically for computer science education workshops.

SIGNIFICANCE OF THIS TOPIC

If the unit proposed here is developed successfully it would be a step toward making our youngest generation more technologically sophisticated. The unit, especially if expanded to include junior and senior high school students, would provide a firm foundation for understanding computers and the thought processes necessary to use them efficiently.

More specifically a unit developed under the above model will have the following advantages:

(1) It can be squeezed into the existing curriculum with a minimum of disruption, it being much easier to make room for a ten or fifteen hour unit than for a term-long course.

(2) Teachers can be trained to use the unit in a fairly

short time. (AND the training will NOT prepare them for higher paying jobs in the private sector, as a unit involving extensive programming and use of computer hardware might).

(3) No large investment in hardware will be required.

(4) Students will be given a brief, yet solid introduction to a powerful type of knowledge - the knowledge of thought processes. This will help prepare them not only to think like computer scientists (when so doing is appropriate, of course), but also making much less vulnerable to the inevitable technological changes of the future. The unit will help produce thinkers and problem solvers instead of students whose only knowledge of computers is so hardware dependent that it will be outdated several years after they acquire it.

SUMMARY

In this paper I have proposed a model for developing a unit to teach the essential problem solving skills of computer science, or at least a reasonable subset thereof, to elementary school children. To ensure validity, faculty in a cooperating computer science department should be interviewed. Activities based upon the information gathered in the interviews and appropriate for elementary school children can then be developed, using as resources the experiences and training of local teachers, the ideas of graduate students in computer science education, and the computer science education articles in various magazines and journals. The activities should be field tested and then taught to teachers in inservice workshops. Finally, these teachers should be observed and interviewed periodically as they attempt to implement what they have learned in the workshop.

The goal here has been to present a model for developing a unit to teach the essential skills of computer science to elementary school students without requiring schools to purchase an inordinate amount of equipment. When finished with the unit students should have better insight into what it means to think like a computer scientist.

Computer science requires specific problem solving skills that can be taught independently of actually using a computer, which is not the same as saying that the learning of these skills cannot be enhanced by providing students with access to computers. As mentioned previously the ideal situation would be to have enough hardware for each student to have access to it whenever the need arose. But it will not be possible in the immediate future to train enough teachers and purchase enough equipment to do the optimal job. Even if it were, however, the content of such a curriculum would require some guidance. The key ideas of computer science would still have to be defined, activities developed and tested, and personnel trained. This paper has outlined a method for attaining these goals.

BIBLIOGRAPHY

BOOKS

Bantock, G. H.; DILEMMAS OF THE CURRICULUM; John Wiley and Sons; New York; 1980.

Cyert, Richard M.; See Tuma and Reif.

Frates, Jeffrey and Moldrup, William; COMPUTERS AND LIFE; Prentice-Hall; Englewood Cliffs, NJ; 1983.

Goodlad, John I. and Klein, M. Frances and Associates; BEHIND THE CLASSROOM DOOR; Charles A. Jones Publishing Company; Worthington, Ohio; 1970.

Goodlad, John I. and Associates; CURRICULUM INQUIRY: The Study of Curriculum Practice; McGraw-Hill Book Company; New York; 1979.

Graham, Neil; THE MIND TOOL; West Publishing Co.; St. Paul; 1983.

Hayes, John R.; See Tuma and Reif.

Hunter, Beverly; AN APPROACH TO INTEGRATING COMPUTER LITERACY INTO THE K - 8 CURRICULUM; Human Resources Research Organization; Alexandria, VA; National Science Foundation; Washington, D.C.; October, 1980;
ED 195 247

Larkin, Jill H.; See Tuma and Reif.

Keating, John et. al.; INTRODUCTORY IN-SERVICE WORKSHOP: COMPUTERS IN EDUCATION; Department of Computer and Information Science; Eugene, OR; 1982.

King, Arthur R., Jr. and Brownell, John A.; THE CURRICULUM AND THE DISCIPLINES OF KNOWLEDGE; Wiley; NY; 1966.

Meyer, Adolphe E.; AN EDUCATIONAL HISTORY OF THE AMERICAN PEOPLE; McGraw-Hill Book Company; New York, Toronto, London; 1957.

Morley, Franklin P.; A MODERN GUIDE TO EFFECTIVE K - 12 CURRICULUM PLANNING; Parker Publishing Company, Inc.; West Nyack, N.Y.; 1973.

Pratt, David; CURRICULUM: Design and Development; Harcourt Brace Jovanovich, Inc.; New York; 1980.

PRISM: PRIORITIES IN SCHOOL MATH: Executive Summary of the PRISM Project; NCTM, Inc.; Reston, VA; 1981.

Robinson, Mary; CAREER EDUCATION MATH: UNITS FOR CAREER EXPLORATION IN SIXTH, SEVENTH, or EIGHTH GRADE; Oklahoma State Department of Vocational and Technical Education; Stillwater, OK; 1974.

Reif, Frederick; See Tuma and Reif.

Rubinstein, Moshe F.; See Tuma and Reif.

Shuster, Albert H. and Ploghoft, Milton E.; THE EMERGING ELEMENTARY CURRICULUM; Third Edition; Charles E. Merrill Publishing Company; Columbus, Ohio; 1977.

Stern, Nancy and Stern, Robert A.; COMPUTERS IN SOCIETY; Prentice-Hall; Englewood Cliffs, NJ; 1983.

Taba, Hilda; CURRICULUM DEVELOPMENT: Theory and Practice; Harcourt, Brace & World, Inc.; New York; 1962.

Taylor, Robert P., editor; THE COMPUTER IN THE SCHOOL: TUTOR, TOOL, TUTEE; Teachers College Press; New York; 1980.

Tuma, D.T. and Reif, F., editors; PROBLEM SOLVING AND EDUCATION: ISSUES IN TEACHING AND RESEARCH; Lawrence Erlbaum Associates, Publishers; Hillsdale, NJ; 1980.

Tyler, Ralph W.; BASIC PRINCIPLES OF CURRICULUM AND INSTRUCTION; U. of Chicago Press; Chicago; 1949.

Weyer, S. A. and Cannara, A. B.; CHILDREN LEARNING COMPUTER PROGRAMMING: EXPERIMENTS WITH LANGUAGE CURRICULA AND PROGRAMMABLE DEVICES; Stanford University Institute for Mathematical Studies in Social Science; Palo Alto; 1974. ED 111 347

Wiles, Jon and Bondi, Joseph Jr.; CURRICULUM PRACTICE: A Guide to Practice; Charles E. Merrill Publishing Company; Columbus, Ohio; 1979.

ARTICLES / INTERVIEWS

"Curriculum '78: Recommendations for the Undergraduate Program in Computer Science"; COMMUNICATIONS OF THE ACM; Vol. 22, No. 3; March 1979.

THE TENNESSEAN; March 13, 1983.

NEWSWEEK; May 9, 1983.

NEW YORK TIMES; "Employment Outlook in High Technology"; March 27, 1983 p. 61.

Schwab, Joseph J.; "The Practical 3: Translating into Curriculum"; SCHOOL REVIEW; August, 1973; 501-522.

Sturdivant, Patricia - Associate Superintendent Houston Independent School District; interviewed in Portland, Oregon; May 12, 1983.

MICROCOMPUTERS IN THE JUNIOR HIGH SCHOOL

Arthur Luehrmann

The title of this conference is apt. For, if the computer is to earn a permanent place in the education of young people, it will do so because parents and educators alike recognize not only the immediate vocational payoff, but also the intellectual dimension of computer skills. If the computer is not seen as an extender of the mind, then it will not be given a place in the curriculum.

I shall attempt to show in this paper that, despite a multitude of competing uses of the computer, the dominant mode of use at the junior high and high school levels is as a medium for the expression of ideas--the ideas being contained in the programs students themselves write. Along the way, students learn the spelling and grammar rules of a programming language and the "survival skills" of interacting with a computer. But those are just means to a higher end, which is the goal of all education: thinking, planning, expressing ideas, analyzing problems, and solving them.

Recent History

During the last three years, we have witnessed a remarkable growth of interest in computers and computing in schools. The number of school computers has grown at an annual rate of between 50 and 100 percent. Computer teacher organizations have sprung up in nearly all states, and membership growth has matched or exceeded that of equipment. Today there are nearly 50,000 members of these organizations nationwide. It is hard to attend an education conference in a traditional subject-matter field without finding special sessions on the computer.

By a head count of computers or teachers, the main activity has in the past been centered in the high schools. Today, however, we are seeing the interest in computers move lower and lower into the school curriculum. Indeed, some software publishers see the preschool market for their products as greater than the school market. Curriculum planners for schools and districts are uncertain when all these trends will settle down into a recognizable pattern.

It seems to me that the settling has already begun and that the junior high school is turning out to be the main place where the majority of students receive their first formal course in computing. As I shall show later, there are good reasons for this, not all of which have to do with the student's age.

This pattern is clouded to some extent by confusion over the many possible ways that computers can be used in education. Some applications are specifically directed at elementary school and others at high school. By looking at the broad spectrum of potential applications, one might mistakenly decide that computers will find the same degree of use at all grade levels.

A Confusion of Purposes

One can penetrate this confusion by classifying the several roles of the computer in learning. Over the years, many people have proposed such classifications. Often they proceed from the point of view of the computer and its capabilities. However, I prefer the simple scheme, origin unknown, based on the learner: The student may learn <u>from</u> the computer, <u>with</u> the computer, or <u>about</u> the computer. These three prepositions seem to cover most of the waterfront.

Learning from the Computer

In this mode, the computer has been programmed to do some or all of the tasks of an ordinary teacher in a traditional subject area. The computer delivers instruction, asks questions, gives prompts, records answers, gives tests, assigns grades, and tracks progress. This application is often called computer-assisted instruction (CAI) or, if only the record-keeping function is used, computer-managed instruction (CMI).

In principle, CAI and CMI have application at any level of the curriculum. In practice, very little software exists for use in topics other than elementary arithmetic facts and operations, spelling, and simple grammar. This is so because these CAI programs are easy to write. They are simple drills in which the examples are generated by algorithm or by lists of paired questions and answers. Once out of this cognitive domain, CAI programs of high quality are enormously difficult to write.

As a result, it is a practical truth today that CAI is of little use in high school or junior high, except for remediation and special education. There are outstanding exceptions to this rule, not the least of which are the college and high-school level dialogs on physics produced by Alfred Bork [1] and his colleagues at the University of California, Irvine. Still, the practical rule given here is right today far more often than it is wrong.

Learning with the Computer

As with CAI, the student in this mode of use interacts with the computer through a program that serves some pedagogic purpose. However, in this mode, the computer is a <u>tool</u>, not a <u>tutor</u>, to borrow from Robert Taylor's terms [2]. The program does not instruct. It does not have a list of questions and correct answers. It does not keep records or give grades. Instead, the computer gives the teacher or the student a tool to use along the way to learning some body of traditional subject matter.

Certainly the most appealing of these tools are simulation programs. They allow one student or the entire class to participate in a game-like activity that is solidly based in the subject matter under study. An example is the Oregon Trail simulation, in which a history student plays the role of a person taking his family west. To be successful, one must make critical decisions about food, water, ammunition, and other things needed for the trip. Another example is the Geography Search program, in which teams of students pilot imaginary sailing ships across an ocean in search of a new continent and a city of gold.

In using simulation programs, the student is put into a problem-solving mode. If the simulation is pedagogically sound, the problems to be solved require knowledge and understanding that is important to the educational goals of the course. Otherwise, the result is merely an abstract game of no particular relevance.

Without doubt, the computer tool with the greatest educational application ahead is the word-processing program. All the early signs are positive: young children who learn to write with word-processing software spend far more time on task, enjoy the experience, become their own critics, and accept more easily the criticism of others. The arrival of this tool is the best news English teachers have had since the graphite pencil.

There are many other computer-based tools with educationally significant applications. They are as varied as the subject matter to which they apply: analyzing and graphing data, solving math problems, simplifying equations, accounting, composing music, drawing pictures, and so on.

In principle, computer-based tools have application to all educational levels and nearly all subjects. However, with the single exception of word-processing software, it seems unlikely to me that we will see rapid or fundamental changes in the content or structure of school subjects as a result of their use. The cause is not so much the quality of the products, some of which are outstanding. Rather, it is the inertia of our system of education, depending as it does on the mutual consent of teachers, publishers, and testing agencies. To introduce a successful change, teachers must agree to alter their approaches and lesson plans; publishers must provide reliable materials that assume computer use as part of the program; and testing agencies must modify items in their tests.

These changes will be slow in coming, despite any obvious benefits of the change, because of the difficulty of getting all these actors to move together. Rational argument is insufficient. If that were enough, schools would have stopped teaching long division as soon as calculator prices hit $10. After all, there is nothing mathematical about long division. It is just a way of getting an answer, and the pocket calculator is a better way. Still, I expect to see long division taught, written about in textbooks, and tested upon in achievement tests for at least fifteen more years.

Word-processing is different. It threatens no established body. It requires no elaborate consensus among teachers, publishers, and testers. The change can happen one school at a time. This is true because the role of the computer is manifestly supportive of traditional educational goals: learning to write and criticize what one has written. The only adaptations required of the school are to provide (1) a little time for students to learn efficient keyboard skills and the use of the word-processing program, and (2) a lot of computers for them to use in their writing tasks throughout school. Declining computer costs will take care of the latter problem.

So I see word-processing programs entering elementary schools and being used at all levels of education. The majority of other tool uses of the computer will take much longer. For the next dozen years or so, they will probably play a role somewhat like that of audio-visual aids to traditional instruction: eagerly used by some teachers, abhorred by others, ignored in textbooks, and unevaluated by the testing agencies.

Learning about the Computer

In the tutor and tool modes of use described above, the computer either delivers or supports instruction in traditional school subjects. The computer does these things by running programs, usually acquired from software publishers.

The mode discussed next is fundamentally different from the earlier two. In this third mode, the computer enters the curriculum as an object of study in its own right. The machine is at the service of a totally new course in the curriculum. The course is called by various names: "Introduction to Computers", "Computer Programming", and "Computer Literacy" are common ones.

Recent data [3] makes it clear that in grades 7 through 12 the dominant use of school computers is in support of courses about the computer itself. If an observer walks into a randomly chosen junior high or high school, locates a computer, and asks the student there what he or she is doing at the keyboard, the most likely answer will be "Writing a program for my computer class."

There are no programs to buy for this third mode of use. The only programs involved are the ones the students in the course write as they learn about the computer. In many respects, the function of the computer in the computer course is the same as the function of the tablet in a writing course. Each one is the object upon which and through which the student expresses ideas, feelings, and understanding.

Robert Taylor calls this the computer's "tutee" mode. The student is the tutor. By writing programs, the student teaches the computer all the steps to go through in solving a problem. To teach the computer, of course, the student must first think hard about the problem and understand all the steps in solving it. Only then can he or she correctly tell the computer what to do and how to do it. All of the intellectual merit of a computer course derives from this simple statement of fact: To write a successful computer program, one must understand the problem and its solution.

As in learning to write English, the student learning to write computer programs must learn a vocabulary, spelling rules, and the grammar of a language. These, however, are but means to a higher goal and not ends in themselves. Programming languages are so trivial, linguistically speaking, that they would not on their own deserve serious attention in the school curriculum. With vocabularies of only a few dozen words, simple rules for spelling and punctuation, and a rigid grammar with no exceptional cases, programming languages are no challenge to anyone.

The challenge comes when one tries to use these simple elements for some purpose, such as having the computer draw a picture, play music, or solve a math problem. These and other tasks cause the student to think carefully about the details of the problem, to consider alternative approaches, to conduct experiments, and to analyze results. All of these things are higher-order mental processes, the exercise of which is the common goal of education in all subject areas.

I shall return to these ideas more fully in the latter part of this paper. The point I want to stress now is the fact that many teachers in many schools with many kinds of students are all reaching the same conclusion: Learning about the computer and learning to control the computer is worth a semester or more of the time of the average student. At the present time, some 25,000 or more U.S. teachers are in the process of becoming a new kind of teacher. Whether formerly math teachers, or English teachers, or librarians, they are now on the way to becoming full-time computer teachers.

The Junior High Years Are Best

Although the interest in such computer-literacy courses spans many years of our educational systems, I see most of the activity in grades 6 through 10. Indeed it is true that many colleges have inaugurated some sort of computer literacy program for their undergraduates. Yet I get a strong sense that these are viewed by all concerned as stop-gap programs, not as permanent parts of the college curriculum. They seem intended to compensate the many students who in high school had no opportunity to learn to use computers. As that situation disappears, so too will the need for these courses.

At the other end of the educational ladder, there are those who would argue that education in using computers should begin in the early years of elementary school. Indeed, the work with Logo suggests that very young children can learn the turtle graphics commands in Logo and can learn to divide large problems into small procedures. While there is a good deal more to Logo, which they do not learn in these early experiences, nevertheless what they have learned is of great fundamental value.

It may be, as some believe, that these small-scale experiments with Logo and other computer languages can be extended to young children everywhere. Not everyone shares that view, however. A recent article by Robert Tinker [4] points out difficulties his group has had in using Logo to teach concepts that go beyond turtle graphics. He notes, for example, that Logo requires a different notation for the name of a variable and the value that the name stands for. The need for this distinction, which is blurred in almost all other languages, is not easy to communicate to beginners, he says.

Yet it is an important distinction. Failure to grasp it is at the root of the confusion over the meaning of the assignment statement X = X + 1. The X on the right means the value of the variable, while the X on the left means the name of the variable. Sooner or later, everyone who writes programs must make the name/value distinction.

It may be, then, that early success in teaching turtle graphics is not necessarily a good indicator that young children are ready for the meat of the language. I suspect that this is the case, at least for average students, and that <u>formal</u> instruction in computing may be vastly more effective when students are a little older and a little farther along in the concrete-to-formal transition.

All of this discussion, however, is purely academic as far as school policies and planning for computer education are concerned. The fact of the matter is that instruction in computing is being borne along today by the willingness of teachers to teach this new subject. While a few third-grade teachers are keenly interested in doing so, the majority of their third-grade teacher colleagues do not share their enthusiasm. So, in deciding what constitutes the third-grade curriculum at their school, disagreements and compromise are likely to ensue.

Contrast that situation with the one that occurs along about the seventh grade, where there are subject-matter teachers, not grade-level teachers. In junior high, one teacher in the whole school may decide that he or she would like to offer one section of a computer course as an elective. That is an easy step for both teacher and school to take. Even if a majority of the other teachers have no interest in the subject, they probably will not raise objections.

I have seen many schools where computer courses began in exactly this way, carried along by the interest, curiosity, and commitment of a single individual. In the first year, there is one section of the course offered. As equipment becomes available and enrolment pressure builds, more sections are added. Before long, the teacher has become a full-time computer teacher and may have recruited a colleague to handle additional sections. By now, discussions will have begun about making the computer course a required one for all students.

This process is also happening at the high schools, of course. But as more and more junior highs take responsibility for the first semester or two of instruction in computing, there will be less need for high schools to be doing the same job. Instead, they will probably begin to offer courses that capitalize on the computer skills already possessed by entering freshmen. These courses include programming applications and computer science.

In summary, the main use of computers in junior high and high school today is in support of courses about the computer itself. For the reasons given above, it seems likely that responsibility for the first formal course in computing will fall quickly to the junior high schools and will probably not fall any lower in the K-12 curriculum.

Computer Literacy is Great—But What is it?

Now I turn again to the question of the content of that first formal course in computing, which many people call "computer literacy". What <u>can</u> be taught? What <u>should</u> be taught?

In the early days of educational computing, all we had for facilities was the time-shared computer system. It consisted of a central computer, costing several hundred thousand dollars, and a few dozen remote terminals. The terminals communicated via telephone lines at typewriter speeds. Although terminals cost only about as much as today's micros, the total bill for computer service and telephone lines came to about $5,000 a year. As a result, it was a rare school that could afford more than a single terminal to some central computer in the district or at a nearby university. Most schools had no computer access at all.

Given the scarcity of computer access, it is easy to guess what learning <u>about</u> the computer meant then. It meant reading books, looking at pictures, and listening to the teacher. If you were a computer teacher in those days, you will remember the effort of putting together a satisfactory computer course in which kids rarely saw or touched an actual keyboard.

The best record of that state of things can be found in the survey work of Johnson <u>et al</u>. [5], which was conducted during 1978-79. The goal of that project was to arrive at a definition of computer literacy by finding out what teachers were actually teaching about computers. They found that the vast majority of precollege computing courses were woefully unprepared to give students significant amounts of hands-on experience. As a result, the courses turned, for the most part, into descriptive, historical, theoretical courses. Apart from learning a new vocabulary and discussing social issues, students emerged from such courses with little ability to do computing.

More than Book Knowledge

Computer literacy is far more than book knowledge. This is not to say that book knowledge is unimportant. Unexamined experience and experience which we lack words to describe is not worth much. I would be the last to suggest that computer literacy is to be achieved by turning 30 kids loose 45 minutes a day in a room full of computers, to do what they want. That would be like expecting ordinary literacy to result from turning kids loose at so many typewriters or tablets of paper, without guidance, instruction, or a vocabulary to describe their efforts. It takes more than experience to learn skills.

On the other hand, it is impossible to acquire any skill, physical or intellectual, without experience. The typewriter and the tablet do not teach one to write; but without them, it is impossible to learn writing skills. These expressive and analytic skills are arrived at through practice, critical evaluation, and revision.

So it has always been with computer skills. The learner must do more than read books, listen to discussions, and build vocabulary. The learner must also try out ideas, give them expression, discover flaws, seek criticism, revise, and try again. It is this ancient process, whether applied in language arts, mathematics, or computing, that finally leads to mastery of skills. Take away this process and the most a teacher can hope for is a general appreciation of the subject at hand.

The goal of all education is higher than this. We do not want our children to become mere appreciators. We want them to become free and independent doers [6]. For this, they need to possess both knowledge and skills.

Thus, as computer equipment has become cheaper and, therefore, more abundant in school settings, teachers have quickly changed the focus, the content, and the structure of their courses. Probably all of us at this conference have watched or taken part in this transition. We have seen what happens when one Model 33 Teletype gives way to a single microcomputer one year, and then three the next, and then eight the following year. We have seen class activities shift from reading and listening to doing and thinking.

But Why Learn Programming?

"Doing" at the computer means writing programs. After all, a computer is designed to do nothing by itself, except follow instructions. So if a student is to have the computer do anything, he or she must tell it what to to. Telling a computer to do something means writing a program. To write a program, one must learn the grammar and spelling rules of a language. One must also learn how to express ideas in the language. That is what [illegible]

[illegible] ple [illegible] forward enough. The [illegible] at [illegible] of a computer literacy [illegible] s sk[illegible] of writing, [illegible] ng, and revising programs. Yet, whenever I draw this "obvious" conclusion, I can sense the hair bristling on the necks of some people in the audience. "Programming" seems to be a dirty word to some people. At least, it seems to them to be an improper subject for the general education of the majority of students in a school. It might do for a few with special interests or vocational needs, but certainly not for everyone.

I think the problem here is a semantic confusion of the words "programming" and "programmer." I would certainly agree that schools have little business preparing the majority of students to become programmers. But I would also point out that schools should not (and do not) prepare most students to become writers or mathematicians. Yet all students should learn to <u>do</u> writing and to <u>do</u> mathematics at a fairly high level of skill while in school. For the same reason, they should learn to <u>do</u> computing. Each of these activities will develop the expressive, analytic, and problem-solving abilities of the student. How these abilities are put to work is a separate issue, largely up to the individual student and the needs of society.

I would not argue strongly for teaching computer language skills if that job were as hard as teaching mathematical skills or language arts. Fortunately, computer languages look like toys when compared to mathematical or natural languages. Computing is certainly <u>not</u> worth 20 to 24 semesters of effort. But it certainly <u>is</u> worth two or three semester-long courses out of a student's 120 semester-course allotment in twelve years of schooling.

Programming in What Language?

Judging from the schools I visit and the reports I read, most teachers of computing today go along with the argument just presented in favor of teaching programming. The backbone of most computing courses in 1983, in contrast to the situation in 1978, is programming and problem-solving on the computer. In fact, the main use of school computers today is in support of programming instruction [3]. The controversy over programming is clearly dying out.

Now a new controversy is on the doorstep: What is the proper computer language to use in the computer literacy course? Although BASIC is the choice of 95% of schools and teachers today [3], that situation is the result of historical accidents and chance decisions by manufacturers of today's micros. Papert [7] argues eloquently against BASIC and in favor of Logo as the correct choice for beginners. Computer scientists at universities generally despise BASIC. They prefer Pascal, by and large. A glance at the want ads, however, will convince anyone that none of these languages is right. To get a programming job, a person seems to have to know either COBOL or FORTRAN.

I find it easy to agree with the critics of BASIC. More correctly, I agree with their critcism of most versions of BASIC as implemented on today's microcomputers. These BASICs look like they were hammered together in a garage by people who did not even bother to read the specifications for the original Dartmouth BASIC of 1965, not to mention the carefully documented revisions published by Dartmouth in the 1970's, nor the documents describing ANSI Standard BASIC. The result has been a Babel of incompatible, tasteless, amateurish, needlessly complex, and often error-laden dialects of what started out as a simple language for teaching computing. The BASICs we have now are awful.

But then, so is English an awful language. It is a hodge podge of Old Norse, Anglo-Saxon, and Norman French. There are practically no spelling rules worth remembering. Its undeclined grammar invites horrible ambiguities of meaning. Even after 16 years of training and practice, few people can say that they have truly mastered the language. Yet, despite these manifest flaws, English remains the language of choice for teaching reading and writing in large regions of the world. Why is that?

Expert linguists have long documented the sorry state of the English language. For many centuries, a knowledge of English was regarded as a necessary evil, while the truly educated would learn to read and write in superior languages such as latin and classical Greek. Experts in the last century have even invented measurably superior alternatives to any of these native languages. Esperanto is the best-known example. Why then, in the face of expert testimony and better alternatives, does English persist?

The answer is simple, it seems to me. The issue of language, <u>per se</u>, is of far less importance than the issue of literate and fluent use of a language. It is not that we want children to learn English because a knowledge of English spelling and grammar is important in itself. Rather, we want children to learn to use English to express ideas and feelings and to understand and appreciate the ideas and feelings expressed by others. English is the choice in Peoria for the same reason that French is the choice in Paris: Like them or not, they are the common tongues in those places.

And, like it or not, BASIC is the common tongue of computing today in schools and homes. Of the 250,000 or more computers in schools today, less than 10,000 are used with Logo or Pascal. Of the 5 million home computers to be sold in the next year, only about ten percent will offer any language other than BASIC, and then only at considerable extra cost to the buyer. Within three years, the vast majority of our students will have direct access at home or at a friend's house to computers that speak BASIC and nothing else. A tiny minority of students will have access outside school to other computer languages.

This circumstance puts computer educators in a tight box. They may feel that another language is better than BASIC for teaching computer literacy and computer fluency. Yet if they choose a language other than BASIC, their students will be unable to practice outside school the things they learn in class. A century of broad-based public education gives solid evidence that the things most students succeed at are the things they get practice and reinforcement in outside school. Hence the demise of Greek and latin in the high-school curriculum.

Making the Most of Micro BASICs

Given these alternatives--shoddy microcomputer BASICs and little nonschool access to better languages--a teacher might become very gloomy. However, as I said earlier, the issue of language is far less important than the issue of literate and fluent use of language to express ideas. We have found in our recent work [8] that it is indeed possible to use these current versions of BASIC, warts and all, to teach the main ideas of computing. These ideas include such "advanced" topics as top-down design and structured programming as well as the usual content of a first course in computing.

By introducing the GOSUB and RETURN statement at the beginning of the course, instead of treating them as advanced topics (as is often done), the teacher can show students how to work on a problem by dealing with it at the appropriate level of abstraction. Students, whether solving computer, math, or writing problems, often get lost in the details and fail to see the big picture. They need guidance. They need to be taught a method of describing the problem in an abstract language that puts off the details until later. In programming problems, this is done by burying details in subroutines or procedures and writing the main routine in terms of references to those subroutines. The main routine is written first and the subroutines come later.

As I have said, this powerful approach to problem solving can be taught effectively in BASIC, though few programming manuals offer any guidance. Programming manuals, for the most part, are the grammar books of computing. Students and teachers also need books on the rhetoric and style of computing.

Structured programming is another topic largely ignored in BASIC manuals. One teaches this method of writing by introducing three fundamental control structures--a straight sequence of actions, a repetitive loop, and a two-way branch--and then showing the student that all problems in program logic can be solved with only these three structures. The result, especially when used with a clear indentation style, leads to programs that are easy to read, easy to get right, and easy to change.

Some critics of BASIC decry the language because, they say, it does not allow structured programming. That is utter nonsense. I grant that the BASIC language permits all kinds of atrocious and unreadable things to be written. So does English. Yet the English teacher manages to coax students into better writing habits. With a little help, so can the computer teacher. There is a systematic way to teach the three standard control structures in BASIC [8, 9]. When students learn it, they are able to write clearly and well.

Finally on this topic, we must remember all those little BASIC-only home computers. We have no choice. Like it or not, there is no way to keep BASIC a secret. What will happen if we fail to teach proper BASIC in school? The result is absolutely predictable: "street-BASIC" users by the millions.

Work Ahead

The arrival in the school of computers and courses in computing has been breathtakingly fast. It has not always been smooth and orderly. A good deal of work remains to be done.

From the point of view of the individual school or district, the number-one need is good planning. It is all too easy when parents and teachers and kids are all clamoring for computers, to rush ahead into equipment acquisition before thinking through a plan of use. The planning issues are too numerous to consider in this paper. They have been discussed more fully elsewhere [10].

Two problems, however, so transcend the concerns of the individual school that they deserve the attention of the whole society. They are discussed below.

Teacher Training

The vast majority of today's computer teachers have moved into this subject area from another one. It is interesting to compare the amount of subject-matter training they have had in their former field with their training in computing.

A math teacher at the junior high or high school level has had 24 semesters of arithmetic and mathematics in school, has taken additional math courses in college and has taken math methods courses a well. An English teacher has had similar amounts of subject-matter preparation.

But what of the computer teacher? There were no computer courses when the teacher was in school. There were probably few or no opportunties to take computer courses in college when the teacher was an education major. In any case, the teacher probably had no thoughts then of becoming a computer teacher. The result is a cadre of about 25 to 50 thousand people in the process of becoming computer teachers, almost none of whom have significant amounts of formal training in the subject they will be teaching.

This problem can be solved, but only by deliberate action of three groups: the keepers of the content domain, the teacher education institutions, and the teachers themselves. The content domain here is computer science. Thus, university computer scientists must involve themselves in preparing school teachers to understand more fully the content of the new subject they are teaching. Schools and departments of education must work work together with computer science departments to plan practical, effective courses and degree programs for teachers. And teachers must seek out institutions that offer such courses and programs.

This is far easier to say than to do. Computer science departments are universally overburdened with the task of teaching their own undergraduate majors. Enrolments grow without bounds and industry attracts the faculty away from the university. Nevertheless, the fact remains that if academic computer scientists shun the job of preparing school teachers of computing, the academics have only themselves to blame if college freshmen continue to appear in the introductory computer science courses with wretched programming habits picked up in school. It is grossly unfair to complain about this problem and at the same time to withdraw from helping out.

It must be stressed that we need solutions mainly for the teachers now in place in schools. Preservice programs should be started as soon as possible, but their impact will not be felt for a decade. In the meantime, we need in-service programs for all those tens of thousands of converted math, English, and science teachers. They are the ones who will be doing the computer teaching for a long time to come.

Equality of Opportunity

The final problem to be addressed here is a profound one for our entire society. The data collected recently [3] makes it clear that children in poorer districts have a far smaller likelihood of using a computer at school than children in wealthy districts. If this pattern is allowed to continue, the consequence will be dire.

The situation is a good deal worse than the survey data indicate, in my opinion. When I visit inner city schools and schools in areas with many migrant workers, I often find more than the usual number of computers. But when I look in on the computer room, what I inevitably see are students at the receiving end of a drill-and-practice program in arithmetic or spelling. These students are the subjects of the computer. When I visit a suburban school, on the other hand, I see students learning mastery over the computer. Both groups have computers, but what they are learning is very different.

I have stressed the intellectual merit of teaching computing as a regular school subject. But one must not overlook the vocational and career impact of computer skills and knowledge. We are, as many observers have pointed out, well into the Information Age. Most jobs and nearly all new ones require that people work with information and information systems. Farm jobs have all but disappeared. Factory jobs are giving way to computer-controlled robots.

The computer is the information machine. As the steam engine amplified a person's power to do physical work, so the computer amplifies ones power to work with information. People who leave our educational systems with knowledge about computers and skills in using them to solve problems will be prepared for work and career growth in the decades ahead. Others, with only manual skills, will face long-term unemployment.

It is essential that members of every community recognize this simple fact and see to it that their children are made ready for this new reality. Education leaders everywhere and at every level must take action to assure that our schools offer an equal opportunity for computer literacy to all our citizens.

REFERENCES

1. Bork, Alfred M., Learning with Computers. Bedford MA: Digital Press, 1981. 286 pages.

2. Taylor, Robert P., The Computer in the School: Tutor, Tool, Tutee. New York: Teacher's College Press, 1980. 274 pages.

3. School Uses of Microcomputers. Newsletter from the Center for Social Organization of Schools, Johns Hopkins University, Baltimore, April 1983. 8 pages.

4. Tinker, Robert. "Logo's Limits: Or Which Language Should We Teach?". Hands-On! Newsletter of Technical Education Research Centers, 6, 1 (Spring 1983), pp. cover-4.

5. Johnson, David C., Ronald E. Anderson, Thomas P. Hansen, and Daniel L. Klassen. "Computer Literacy--What Is It?" Mathematics Teacher, 73 (February 1980), pp. 91-96.

6. Luehrmann, Arthur. "Computer Literacy--What Should It Be?" Mathematics Teacher, 74 (December 1981), pp. 682-686.

7. Papert, Seymour, Mindstorms: Children, Computers, and Powerful Ideas. New York: Basic Books, 1981.

8. Luehrmann, Arthur and Herbert Peckham, Computer Literacy--A Hands-On Aproach. New York: McGraw-Hill Book Company, 1983. 370 pages. Separate versions for Apple and TRS-80 computers.

9. Luehrmann, Arthur. "Slicing Through Spaghetti Code". <u>The Computing Teacher</u>, **10**, 8 (April, 1983), pp. 9-15.

10. Luehrmann, Arthur. "Planning for Computer Education--Problems and Opportunities for Administrators". <u>NASSP Bulletin</u>, **65**, 444 (April, 1981), pp. 62-69.

USING COMPUTERS IN SECONDARY BUSINESS CURRICULUM

DORIS K. LIDTKE

Abstract

Computers are an essential part of both everyday life and business practice of today's world. It is essential that students acquire appropriate skills with an understanding of computing to enable them to function effectively in our computerized society. Incorporating microcomputers into the business curriculum will allow students to acquire the needed skills and understanding. Further, the use of microcomputers in the classroom can have a positive motivational effect for students, provide a means for individualization of instruction, and introduce practical preparation for employment.

Need for Computer Literacy

Educators, parents and employers all agree that students should be computer literate. Professor Preston Hammer stated in the keynote address of the first Special Interest Group on Computer Science Education (SIGCSE) Symposium in 1970 in Houston, Texas, that we teach in elementary school those things that everyone must know, in high school those things that many people need to know, in college those things that some people need to know, and in graduate school those things that only specialists need to know. Hammer continued by saying that computer programming would probably be taught in the fifth grade, since nearly everyone would need to know the fundamental concepts of programming, primarily to understand how a computer works. Hammer was far-sighted. He had a vision of the importance of computers and their uses. Today most educators fundamentally agree with what Hammer said years ago -- nearly everyone needs to understand about computers and needs to be able to interact comfortably with computers in order to function effectively in our society. In other words, computer literacy has become a necessity. Perhaps we should say that knowing about and being able to use computers is essential if a person is to be literate.

There is, nevertheless, controversy over the exact meaning of computer literacy. The spectrum of definitions ranges from the idea of a simple awareness of how computers work to the demand for competency in computer programming. Some middle ground seems to be the most rational current position. To be computer literate students should understand what computers are, how computers function, what computers can do, the societal impact of computers, and be able to demonstrate that they, the students, can use a computer to accomplish useful tasks. The level of expertise the student should attain will increase as the student matures in school.

Historically computers have been viewed as devices to process numeric data. They have been viewed as number crunchers and people with math anxiety have frequently exhibited computerphobia. This should not be true. Computers are information processors. The information may be alphabetic or numeric. The computer is equally adept at processing words, symbols, and numbers. It is important that students understand this.

Computers In Business

While computers have been used extensively in some aspects of business for several decades, the recent widespread use of word processors and microcomputers means that computers have become essential in all phases of business. From small one-person businesses to large conglomerates, computers are used by nearly everyone. From clerk to company president, computers are used for data entry, information retrieval, decision making, payroll, inventory, accounts payable and receiveable, forecasting, planning, data capture and analysis, automatic reordering, automatic telephone dialing and answering, monitoring and control, word processing, and graphics. The student with an ability to use computers will be prepared to live in the information age, whether he/she goes on to post-secondary education or enters the job market.

Though many companies develop their own software, the majority of computer users can rely on available software packages. They need only know how to use particular software packages which accomplish the necessary/desired tasks. This level of usage is important for every business employee. At the next level it is desirable for users of software to have some expertise in choosing which package best meets their needs. These two important interactions with computers, using and choosing software, are, and probably will remain, essential job skills for the next several decades.

Computers In The Business Classroom

There are several software packages available which can enhance the business classroom. These include packages for 1) teaching and improving key-boarding skills, 2) word processing, 3) practical business applications, 4) payroll, 5) accounting, 6) electronic spreadsheets, 7) trend analysis, and 8) graphics. These software packages can be used to enhance the curriculum through better student motivation, spending more time on concepts and less on busy-work calculation, and dealing with more realistic applications. The computer makes larger problems more practical than when all calculations must be done by hand.

Keyboarding Skills

Many software packages are now available to teach keyboarding skills. There are several variations (1), those like Mastertype (2) which teaches the use of the keyboard and Typing Tutor (3) which builds speed and accuracy. These packages may be used to allow students to work at their own pace and/or as a supplement to the regular classroom instruction.

Some complete courses in keyboarding are also available. An example of such courseware is the material from South-Western Publishing Co. (4) which includes four diskettes with software, a textbook, and audiocassette tapes. This is designed as fifty lessons to be used as a ten week term of study.

Whether separate software packages or the complete courseware package are used, both support use of a microcomputer keyboard and an electric typewriter. This allows implementation of these computerbased packages without necessitating one microcomputer per student, which is generally not available in schools today. It seems practical to gradually replace some of the electric typewriters with inexpensive microcomputers and monitors. To learn and improve keyboarding skills would not require a printer.

Word Processing

Some secondary schools may wish to use word processing machines. For many secondary schools the use of word processing software with microcomputers is a practical means to give students access to word processing. Whichever means is used, it is important for business students to become familiar with word processing.

There are many word processing packages on the market today (5). Until recently these packages were quite expensive. Currently several good packages are relatively inexpensive. One such package was especially designed for use in schools, the Bank Street word processor. It is not easy to recommend a word processing package because users have different needs and various reactions to the options provided. Whichever package is chosen, it is important that students become aware of the variety of word processors, their strengths and weaknesses. It might be appropriate to have students read about word processing packages and/or visit computer stores and write a report on the similarities, differences, costs and/or special features of several packages.

Some discussion or demonstration of the special features of word processors and the packages used in conjunction with word processors should be considered. Formatting, multiple copies, underlining, pagination, boldface, subscripts, mailing lists, merging files and labels, different type fonts, and form letters are examples of these features. Students should be made aware of spelling correctors. Whether or not students should have access to or use spelling corrector packages is probably a very controversial issue.

Practical Business Problems

Students need to develop ease in dealing with practical problems such as simple and compound interest, installment purchases, loans, payroll and simple tax computation. Since most businesses, large and small, now use computers for these applications, it seems appropriate to teach students to solve these problems with a computer. The Minnesota Educational Computing Consortium (MECC) (6) has prepared materials in this area for use in secondary schools. These materials include the necessary software, instructional guides, and reproducible student worksheets. These make a fine supplement to the regular curriculum. The Payroll and the Accounting software allow the students to gain valuable, realistic experience by working through projects for a hypothetical company. (See Appendix A. The description of these programs from the MECC Catalog). Some of this material is appropriate for a business math course, and some for a two-year course in Accounting. These materials are designed for grades 9-12.

Electronic Spreadsheets and Their Extensions (7)

With the development of electronic spreadsheets, such as Visicalc (TM), businesses have access to a powerful tool. Students can use these tools in many modes. They can learn to use them to 1) prepare simple reports designed by others, 2) design simple reports, and 3) design reports for others to complete. The students can realize the scope and power of this tool, especially if they are given realistic problems to solve. The students can even learn to use the electronic spreadsheet as a simple word processor.

At the present time electronic spreadsheet software is expensive. Fortunately at least some software companies are willing to give substantial discounts for educational use of their products.

The data files created to use an electronic spreadsheet can then be used for trend analysis and for the preparation of graphic reports, pie charts, bar graphs and line graphs. It is important for students to see how easily these tasks are performed and the quality of the results.

To obtain the full advantage of electronic spreadsheets and their extensions a printer is needed. It is not possible to display the entire report or graph on a monitor.

Summary

This brief presentation has covered some of the ways in which currently available software can be used in the business curriculum in secondary schools. New software is being developed, and appears regularly. It is important for teachers to be aware of new products. Three good sources are Creative Computing, The Computing Teacher and information from the Minnesota Educational Computing Consortium.

There is also a need for study of computers and computing in the secondary business curriculum. This would probably be best addressed by a course called Introduction to Computing. Students should learn about the hardware and software of computers, how computers affect business and society, careers in computing, what computers can do, how computers do their work, and how to prepare simple programs. With this background secondary students will be prepared to use computers comfortably in their daily life, to use computers in their work, and to learn more about computers and computing as the need arises in their future careers.

References:

(1) Starnes, Stephen "Learn to Touch Type." Creative Computing, (April, 1983), pp. 210, 212.
(2) Available for Apple II, Atari and IBM PC
(3) Starnes, pp. 210, 212.
(4) Crawford, T. James, Lawrence W. Erickson, Lee R. Beaumont, Jerry W. Robinson, and Arnola C. Ownby, Basic Information Keyboarding Skills, Cincinnati, Ohio: South-Western Publishing Co., 1983.
(5) Many of the word processing packages are reviewed in Creative Computing, June, 1983.
(6) Minnesota Educational Computing Consortium, 2520 Broadway Drive, St. Paul, MN 55113-5199.
(7) Beil, Donald H. The VISICALC Book, Apple Edition, Reston, VA: Reston Publishing Co., Inc., 1982.

APPLE

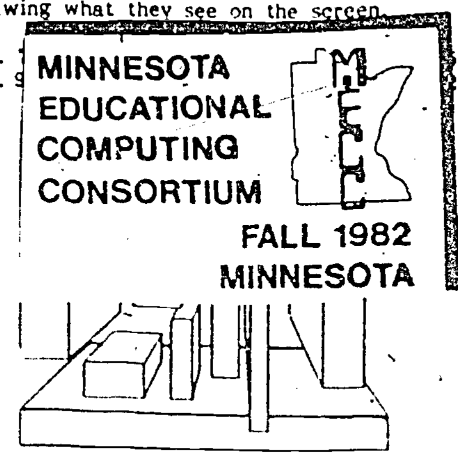
MECC Instructional Computing Catalog

ART

(Grades 7-12)

...paced instruction, Art-... introduces the study of one-... perspective. Problems are ... terminology is explained through ...aphics. Handouts in the support ...re coded to appropriate sections of the ...uter programs to assist students in ...ompleting the problems and reviewing and drawing what they see on the screen.

No. ... .00
No. 9... .00



BUSINESS EDUCATION

BUSINESS VOLUME 1

(Senior High-Adult)

A collection of very practical programs, useful in a number of business education settings, is found in Business Volume 1. Some topics covered by the tutorials and problem-solving programs on the diskette are: interest on installment buying, effects of simple and compound interest, loan amortization, business financial reports, money supply, payroll, and the 1040A tax form. Support materials provide worksheets and input forms for use by the students.

No. 714 Apple II 32K Diskette $15.00
No. 914 Support Manual (46 pages) . . . . $ 2.50

BUSINESS VOLUME 2 - PAYROLL

(Grades 9-12)

Most offices use the computer and today's business education courses must help students gain a broad knowledge of these computerized business operations.

Business Volume 2 contains a set of information processing programs that allow a class to set up and experiment with a computer-based payroll system. These programs and support materials provide experience in creating data files for a hypothetical company payroll. They also allow the student to process the payroll, print checks, and prepare W-2 forms and quarterly reports.

Equipment requirements include dual disk drive and a printer.

No. 713 Apple II 32K Diskette $15.00
No. 913 Support Manual (64 pages) . . . $ 3.00

BUSINESS VOLUME 3 - ACCOUNTING

(Grades 9-12)

The computerized office is here to stay. Accounting tasks are being done on computers and the business education classroom is responding by preparing students to use this new "electronic tool." Business Volume 3 provides realistic experience with a computerized accounting system. It is designed to supplement an advanced or second-year accounting class. The package consists of four integrated systems: general ledger, accounts payable, accounts receivable, and inventory control.

Business Volume 3 contains a prepared set of account vendors, outstanding vouchers, customers, outstanding invoices, and inventory items for a hypothetical company. Students use a series of projects and processing cycles from the support manual as they learn the four systems. An accompanying answer key contains test answers, correctly completed input forms for each project, and all reports and proofs completed by students as part of the projects.

Equipment requirements include a dual disk drive and a 132-character printer.

No. 721 Apple II 32K Diskette $15.00
No. 921 Support Manual (144 pages) . . . . $ 7.50
No. 923 Answer Key . . . . $ 4.50

NEW DIRECTIONS FOR COMPUTER COURSEWARE

J. D. Fletcher

WICAT Education Institute

Perhaps we can begin by agreeing that the ultimate in instructional technology is a log. This notion may come as a surprise to readers elsewhere in the country, but it seems safe to assume that the virtues of logs will be familiar to an audience in the Northwest. As is true of most technology, the effectiveness of the log depends, to some extent, on how it is used. In the case of instruction, we must put one, and only one, student on one end of the log and Mark Hopkins, Louis Agassiz, or any other master tutor on the other. The resulting instructional setting is nearly ideal. Of course, and like all technology, the log is not a panacea. Additional support such as texts, laboratory materials and equipment, and perhaps even filmstrips and computers may also come in handy, but the log properly used in the manner described goes a long way toward solving our instructional problems.

The difficulty with this technology is that we cannot afford it. We cannot afford to put just one student on one end of the log. In view of current educational practice the number is more like 25 or 30. The other end of the log presents some difficulty as well. There are not enough master tutors to go around, and many of these who might be available have chosen to pursue careers in fields more financially rewarding than teaching. Unfortunately, therefore, we must abandon the log as a practicable instructional technology, ideal as it may be. We must return to our ordinary classroom with many students and a single teacher and try to fill the gap between the real and the ideal as well as we can.

However, before abandoning logs entirely it may be well to ask what it is that makes them such excellent instructional technology. As might be expected, much depends on the end of the log concerned with teaching. It must posses three capabilities: subject matter expertise, a capacity for modeling student knowledge, and a complete set of tutorial tactics and strategies.

First, in other words, the teaching end of the log must incorporate at least the level of subject matter expertise that the student is expected to attain. In fact, it may require more subject matter expertise than that targeted for the student, but, at the minimum, it should represent the ideal to which the student aspires.

However, we all know people who are subject matter experts but terrible teachers. Subject matter expertise seems to be a necessary but insufficient capability. This insufficiency may stem from the inability of an expert to represent or "understand" the students' incomplete, inaccurate, and/or contradictory view of the subject matter. Second, then, the teaching end of the log must be able to model what the student knows about the subject matter to be taught. Missing information is not so much a problem here as inaccurate and contradictory information. The expert can probably use his knowledge to represent what information the student needs by shading in areas the student has mastered and emphasizing what is left. However, the expert, precisely because he or she is an expert, may have few, if any, ways of representing inaccurate and/or contradictory information. It is this type of information, so puzzling to the expert and so familiar to the classroom teacher, that blocks many students from subject matter mastery.

A simple, if not trivial, pattern may be emerging from this brief catalog of tutorial capabilities. The teaching end of the log must be cognizant of where we want to go (the model of expertise), where we are now (the model of the student), and now only lacks the knowledge of how to get from here to there. The third capability that must be possessed by the teaching end of the log is that of an expert tutor. It must be able to apply the appropriate instructional tactics at the appropriate time. This capability implies the presence both of a large repertoire of instructional tactics and a strategic understanding of how best to use them. This third capability seems to come most prominantly to mind when we consider the practice of master teachers. Subject matter expertise and knowledge of the student are pushed aside as we consider critical moments when great teaching suddenly illuminates vast new areas of the subject matter through the application of exactly the right instructional tactics at exactly the right time.

In addition to these three capabilities there are two features of the log, as an example of ideal instructional technology, that are worth noting. Again assuming it is used properly, the log will be both individualizing and interactive. With regard to the former, individualization, it is notable that the teaching end of the log can adjust instructional pace, content, media, and methods all in response to the unique needs of the student on the other end of the log and all with the intent of maximizing his or her progress through the subject matter. The second feature, interaction, is in one respect a form of individualization--whenever the student choses to make a response, the teacher will respond back. In another respect, however, interaction is important in its own right because in the tutorial dialogue the student must act. He cannot simply sit passively waiting for information to drift his way. All great tutors insist on the active participation of their students, and those employing the log technology will be no exception.

It should be noted that the American classroom teacher possess most if not all of these attributes. Certainly most classroom teachers possess a level of subject matter expertise sufficient for their students' needs, they can model students' understanding of the subject matter, they possess a repertoire of tutorial tactics, and they generally know when to apply these tactics. Moreover they can individualize instruction and teach in a fully interactive mode provided the student-teacher ratio does not greatly exceed 1:1. One might even argue that the log featured in the discussion to this point is superfluous. This, however, is a possibility outside the scope of this paper and will be left to future scholarly debate.

The problem, again, is that we can no more afford 1:1 student-teacher ratios in our schools than we can the log technology discussed earlier. This seems to be where computers come in. We know the ideal--an Aristotle for every Alexander as Suppes and Morningstar suggested in 1969--, and we know the real--a student-teacher ratio of 20:1 in our elementary schools and 17:1 in our high schools (Dearman and Plisko, 1980). What we are trying to do with computers is make the real a little more like the ideal. This seems a trivial point, but it is frequently lost in the rhetoric surrounding the use of computers in instruction. We are not trying to revolutionize education, provide a personal computer for every student, harness the impressive capacities of the electronic revolution, or hasten us all into the information age. All of these may be useful, secondary gains, but at base what we are trying to do is improve instruction. If the computer helps, so much the better. If the computer fails to help, so be it.

The essential question then seems to be: Are computers adequate substitutes for logs? Or, for those who want to argue for the superfluity of logs: Can computers in the classroom compensate for our inability to afford a full-time master tutor for every student? The answer is a firm yes and no. As we begin to address the subject of new directions for computer curriculum, it is possible to discern ways in which the computer promises to fill the gap between the real and the ideal. It is also possible to discern entirely new "functionalities" in instruction that come to us by way of computers. As is true of most technological efforts, we begin by trying to enhance the capability of our existing practice and end with new capabilities that change the nature of what we do in ways completely unanticipated and unexpected at the beginning. This may be the essence of the computer revolution in schools. It is not just that we will have computers everywhere or that we will enhance our capabilities to instruct, we may also change our ideas about what instruction is. Not only may we get better at doing what we do now, but in a fundamental sense we may change what it is we want to do.

We can therefore expect to see two themes of development in our new directions for computer curriculum. One theme is concerned with replacing the log. That is to say that we will be trying to achieve through computers our current ideal of one master tutor for every student. The other theme concerns the development of whole new capabilities of instruction heretofore unanticipated and unenvisioned. Both of these themes can be seen in the current development of curriculum for computer presentation.

New Directions

There are many ways to categorize what we do when we apply computers to the problems and processes of instruction. Becker's (1983) well balanced discussion of microcomputers in the classroom presents as good a range of categories as any. Becker discusses six categories of instructional uses of computers: drill-and-practice, tutorial dialogue, simulation and model building, management of instruction, teaching computer-related information skills, and teaching computer programming. The fourth of these concerns, as it says, management of instruction, and the last two of these seem to be concerned with a specific subject matter, perhaps best described as computer literacy. Since the intent here is to focus on how things in general are taught rather than on management or on specific computer engendered subjects, subsequent comments will concern only the first three. Also, by way of disclaimer, treatment of these three is not intended to be a comprehensive "snapshot" of the state-of-the-art. Instead the emphasis is on new directions for instructional use of computers both to enhance achievement of current instructional goals and to go beyond them.

It may be well to begin with a fable. This fable is apocryphal--or it would not be a fable--and it may already be familiar to the reader. Nevertheless, it seems sufficiently relevant to bear repeating. As the story goes, there once was a governmental "blue-ribbon" commission of instructional experts assembled to specify the ultimate in instructional technology. After several days of meetings--suitably fueled by long lunches and accommodated by comfortable lodging--the experts came up with the following specifications for the new technology:

* There should be no exotic power requirements. The technology should use ordinary household current, or better be battery powered, or at best require no power at all to operate.

* It should be light and easily portable. One person should be able to transport it, and at best it would be carried in one hand.

* There should be no complicated installation or environmental requirements. It should be easy to set up and use, it should operate in moderately extended temperature ranges, and it should be, as the military says, "ruggedized."

* It should provide random access to a large amount of material.

* It should be capable of displaying graphics, photographics, drawings, color, and high-quality, easily readable text.

* It should be inexpensive, costing less than $50 a copy.

The commission report was received with great relief for, as the perspicacious reader may realize, no research and development money was required to develop the technology. In fact, the technology already existed and had been in place for over five hundred years. The appropriate technology was, of course, a book.

This is a fable for all of us in the business of applying new technology to instruction. We must come up with solutions that promise real innovations--in the case of instructional technology they must be better than books. At the same time some of our prototypes will be--like the horseless carriage--less efficient than what they are intended to replace. Printing was important because, among other things, it was able to capture instructional content and make it inexpensively available to an unlimited audience. As Bunderson (1981) pointed out, computer technology is important because, among other things, it makes both the content and interactions of great instruction inexpensively available to an unlimited audience. This promise has yet to be realized, but it seems almost inevitable in coming. What we need to do is sift through all the prototypal development and find therein those embryonic techniques that promise to be better than books--and as good as or even better than a log. It turns out that these techniques are neither easy to find nor trivial to develop. We will briefly examine them in the three areas of drill-and-practice, tutorial dialogue, and simulation.

Drill and Practice

'Drill and practice' is doubtless one of the more regretable terms in instruction, evoking images of the classroom as a sweat-shop and attracting the ire of those who want to use computers to create a rich and friendly learning environment for intellectual exploration and discovery in the classroom. Certainly it is now fashionable to deprecate drill and practice as a computer instruction technique, and it has probably been so for the last five years. Papert (1980) cites drill and practice as an example of the QWERTY phenomenon. It turns out that because the mechanical keyboards of earlier times were unable to keep up with skilled typists--the keyboards would jam and otherwise misbehave if they were operated too quickly--typewriter keyboards were originally designed to slow down the key presses of skilled typists. The result was the QWERTY keyboard, named after the topmost row of letters. This keyboard is still with us today despite our having removed all the mechanical obstacles to fast operation that resulted in the QWERTY design in the first place.

Papert's argument is that early applications of computers to instruction necessarily followed drill and practice formats partly because that is what classroom teachers would accept and partly because the computer technology of earlier times could support nothing else. This point of view is not entirely accurate as can be seen in the design of curriculums for the IBM 1500 System in the mid-1960s. The Stanford beginning reading program is a case in point. This curriculum, which was designed roughly in the period 1964-1966 and was described more fully by Fletcher (1979), encouraged children to build matrices using words and spelling patterns, read and be read stories (with illustrations), record and play back messages, and experiment with linguistic forms and variations. Teacher acceptance was an issue somewhat separate from the content and approach of the curriculum--using computers to teach at all and taking away from classroom time to do it were the central concerns of the teachers. Nonetheless it is notable that when the Stanford group moved to a less expensive machine configuration for presenting beginning reading instruction, the curriculum design became more drill and practice in nature.

In any event, it seems past time to make a few arguments in favor of drill and practice--reckless and dangerous though it may be to do so. Is drill and practice an example of Papert's QWERTY phenomenon? The answer seems to be "no," partly because it works--drill and practice is still one of the most successful techniques we have in computer instruction--and partly because there is so much yet to be tried and developed in the drill and practice mode. Even if we assume drill and practice is limited to presentation of discrete items such as "math facts" or vocabulary items to students, there are at least three directions for curriculum development in drill and practice. These have to do with performance goals, optimal item selection, and optimal activity selection.

Performance Goals. There is something in the world called trajectory theory. Basically this "theory" is a way of accounting for the progress, or "trajectory," of individual students through a curriculum as a function of the amount of time they spend working in the curriculum. Figure 1 shows, perhaps more clearly, what trajectory theory is getting at. For individual students A, B, and C we try to predict and prescribe their grade placement on standardized paper and pencil tests based on the amount of time they spend on the computer curriculum. The interesting thing about trajectory theory is not just that it works, but that it has worked amazingly well in practice. In two published studies using trajectory theory (Suppes, Fletcher, and Zanotti, 1975 and 1976) the standard error of estimated grade placement was in the range .04-.06 of a grade placement. In other words, the estimates were off by less than a tenth of a grade placement for 90% of the cases. Again, these estimates were based solely on the amount of time the student spent on the computer and completely separately from what was being done in the classroom. If we want to predict and control progress toward measured goals of achievement then trajectory theory may be one of the best techniques we have. It is worth emphasizing that although trajectory theory was developed for drill and practice, it may be applied to any form of instruction where we have closely watched and accurate measures of time on task--such as we have in computer instruction.

There are still many questions to be answered about trajectory theory. Can it be applied to all subject matter? Can it be applied to methods of instruction other than drill and practice? Are there significant and important benefits to be gained from using classroom observations of time on task as well as computer time to predict and control progress? The list of questions could be continued. Trajectory theory is not a particularly new technique for computer curriculum, but it remains promising and worthy of further development.

Optimal Item Selection. There is probably no more certain way to put an audience to sleep than to begin discussing applications of optimal control theory to instruction. Accordingly this section on optimal item selection and the next on optimal content selection will be kept mercifully short. However, these approaches do seem to be important and worthy of brief mention.

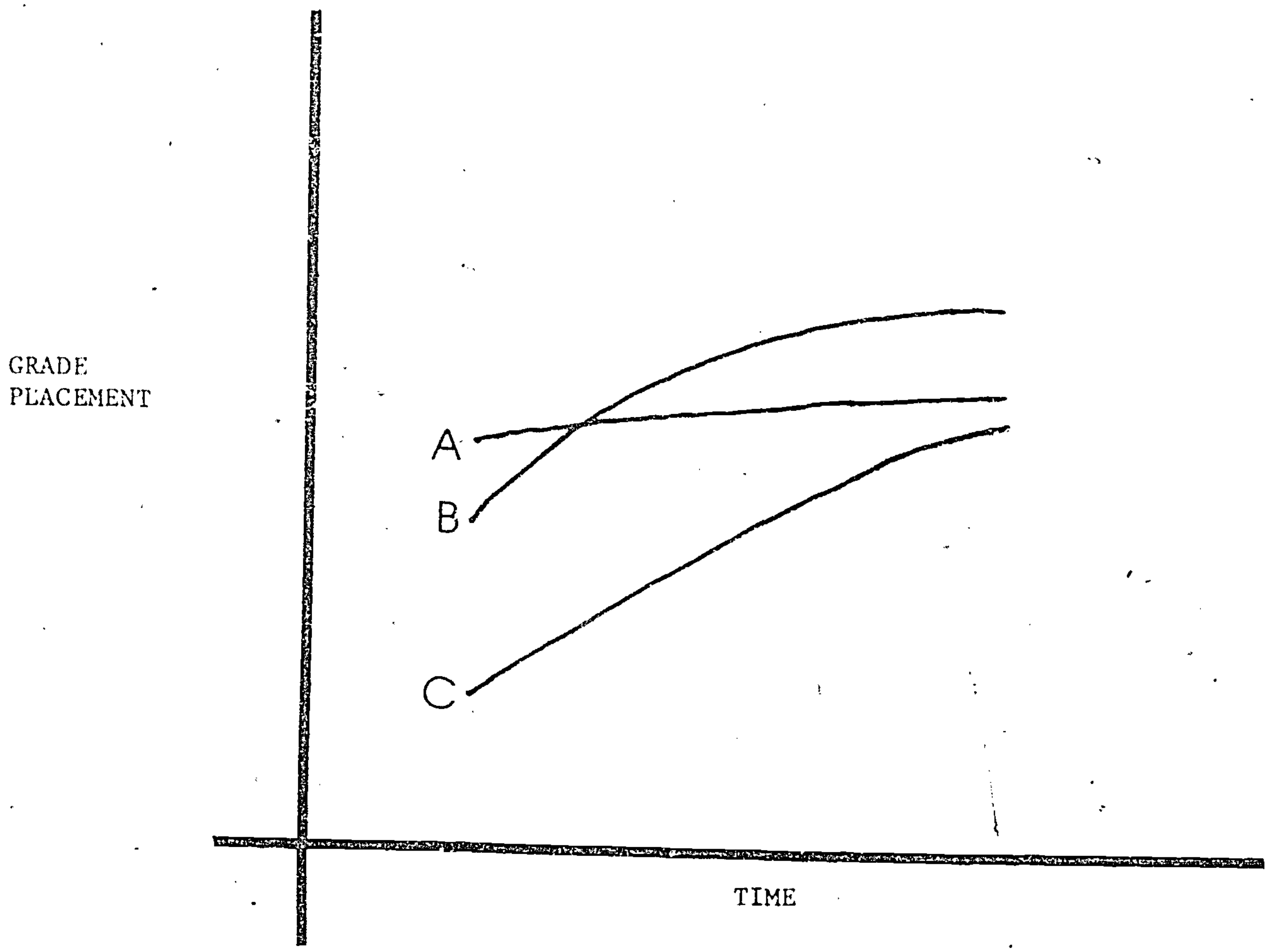


Figure 1. "Trajectories" of students through a curriculum.

Basically an optimal solution is one that attempts to maximize some outcome, such as scores on an achievement test, subject to some constraints, such as total time on task, session length, and student ability. Optimal solutions are brought to us by control theory which, in turn, comes from operations research. It is a well known and noted fact that operations researchers tend to attack problems by removing from them everything difficult to quantify or measure and building an imposing mathematical structure on what is left. In the current instances, the imposing mathematical structures remain, but some portion of what is difficult to quantify or measure can be supplied by mathematical models of learning and memory. The wherewithal for applying both these models and control theory to instruction in "real-time" is provided by computers in the context of computer instruction.

The problem of optimal item selection in instruction was stated with appropriate mathematical clarity and rigor by Suppes almost 20 years ago (Suppes, 1964), but it can be stated fairly simply in words: given a large number of items to be taught and a fixed time to teach them, what subset of items should be presented to an individual student at a given time in order to maximize his or her eventual achievement? The answer can be supplied by the above mentioned quantitative models of learning and memory. Figure 2 presents a probability state-transition matrix of an appropriate sort based on General Forgetting Theory (Rumelhart, 1967; Paulson, 1973). This matrix shows what can happen to an item when it is presented to a student. As can be seen from the figure, the model of learning postulated is very simple. If an item is in the learned state, it stays there. If it is in the short-term state, it will either advance to the learned state or stay where it is. If it is in the unlearned state it can advance either to the short-term state or the learned state or remain unlearned. General Forgetting Theory is actually a little more sophisticated than this in that it accounts for probabilities of correct responding separate from the learning status of items and, notably, it postulates what happens to the learning status of an item when it is <u>not</u> presented. An optimal strategy for item selection based on General Forgetting Theory is, like all models of this sort, fairly simple in its view of human learning but fairly complex to implement. It could not be implemented in a teaching strategy by a book--or by a teacher on the end of a log.

Studies by Lorton (1973) for teaching spelling and by Laubsch (1970) for teaching foreign language vocabulary have shown approaches of this sort to be effective. They may even be dramatically effective, far more so than any other method for teaching large numbers of relatively independent items to students, but little work has been done on them since the mid-1970s. It seems to be a thread of research we have let slip through the cracks. There seems to be no real reason to drop it from our list of new directions for computer curriculum. Its promise of exceedingly efficient instruction remains.

<u>Optimal Activity Selection</u>. A few words may also be in order for optimal selection of activity. This problem most clearly emerges in the context of "strands" approaches to curriculum development. The

LEARNING STATE
AT TIME T+1

$$
\begin{array}{ll}
 & \begin{array}{ccc} \text{LEARNED} & \text{SHORT-TERM} & \text{UNLEARNED} \end{array} \\
\begin{array}{l} \text{LEARNING STATE AT TIME T} \end{array}
\begin{array}{l} \text{LEARNED} \\ \text{SHORT-TERM} \\ \text{UNLEARNED} \end{array}
 & \left[\begin{array}{ccc} 1 & 0 & 0 \\ c & 1-c & 0 \\ a & b & 1-a-b \end{array} \right]
\end{array}
$$

Figure 2. Probabilities of an item state transition when it is presented at time T.

strands approach, which was first described by Suppes (1967), calls for the apportioning of a computer curriculum into various content areas--or strands. For instance, a curriculum in reading comprehension might be divided up into vocabulary, literal comprehension, and interpretive comprehension strands. The problem then for a computer curriculum designer is to decide how much time students should spend in each strand, or, to state it a little more completely, how to control student progress in each strand so that each student's achievement is maximized at the end of some specified period of time. If progress in each strand is independent of progress in each of the others and if each of the strands contributes equally to the measure of achievement, then the solution is simple--we just pick [illegible] and in which learning rate is greatest and allocate all the [illegible]'s time to it. If, however, the situation resembles our [illegible] comprehension example in which progress in one strand is interrelated with progress in the others, the situation is more complex. In reading, after all, a student with a poor vocabulary will not progress very far in literal or interpretive comprehension, yet the achievement measure of success for the curriculum will presumably be more concerned with comprehension than with vocabulary growth. Some sort of optimal mix of vocabulary development and work in comprehension will have to be devised for the student.

An appropriate optimal strategy (based on the Pontryagin Maximum Principle of control theory--for those who are interested) for adjusting progress in interrelated strands was devised by Chant and Atkinson (1973) for the case of two strands. This strategy determines how much time a student should spend each day in each strand depending both on the student's learning rate in each strand and on how much he or she has progressed already in the strand. Extension of the strategy to curriculum environments with three or more strands was left by Chant and Atkinson as an exercise for the reader, but was described by the authors as being "straightforward." It very probably is, but it has [illegible] done, or least published. Moreover [illegible] have been no applications of this strategy to determine in practice how much it really buys in terms of student achievement relative to other approaches. In other words, here is another promising direction which we have just begun to explore. It cannot be implemented in a book--or on a log--and more needs to be done.

Most experimental psychologists reading the above discussion of drill and practice computer instruction will find it difficult to suppress dark and uncomplimentary mutterings about "1960's psychology." There seem to be cycles in research, as in most things. In this dimension we seem to oscillate between attacking small, tightly constrained, and fairly uninteresting problems over which we exercise a great deal of control and attacking very large, sloppy, and interesting problems over which we can exert very little control. As may be evident from the above discussion and in reviews by Atkinson and Paulson (1972) and Fletcher (1975), drill and practice emphasizes the former. Nonetheless, it should also be evident that drill and practice is not just a [illegible] throwing items at students who are [illegible]ted in some assembly line fashion. There are deep, educationally significant, and scientifically credible issues yet to be settled concerning drill and practice. Finally, it should be evident that despite the early strong results we have had from drill and practice much more could be done to fully realize the promise of this approach.

As far as the oscillation between tightly controlled less interesting problems and poorly controlled but much more interesting problems is concerned, it appears that current research in psychology, applied psychology, and instruction emphasizes the latter. This trend is especially apparent in current attempts to build tutorial dialogue systems. Nowhere is the attempt to automate single tutor/single student dialogue more evident. This is the line of development to turn to next.

Tutorial Dialogues

Before diving into the area of tutorial dialogues, a few comments on the automation of programmed textbooks may be in order. Many commentators on tutorial dialogue approaches include in this category the intrinsic programming techniques of Crowder (1959) that appear so frequently in commercially available computer instruction materials. Basically this approach uses the computer as a very large and sometimes very intricate programmed textbook. This is an approach that could be pursued in a book--although the book might have to be carried around in a wheelbarrow. Nonetheless, this approach appears to concern application of book and text technology rather than computer technology to instruction. It remains one of the most common, easily produced, and frequently implemented approaches, and it is best supported by authoring languages for computer instruction. The development of authoring languages such as PILOT, TUTOR, WISE, PLANIT, etc. all seem to have intrinsic programming in mind since this is the approach most easily [illegible]en when one uses these languages.

We tend not to publish our unsuccessful attempts at computer instruction, among other things, but there seems to be an underground consensus among those in the business that these intrinsic programming approaches do not work very well. What appear to be intuitively obvious and correct procedures for assessing student knowledge, deciding when to branch, and providing remedial and accelerated material turn out to be relatively ineffectual in the light of student performance data. The determined reader is welcome to peruse Fletcher and Beard (1973) as an example of unpublished--and unsuccessful--work of this sort. In any case, this section will not concern the automation of programmed textbooks.

This section is concerned with the development of intelligent tutorial systems as a new direction for computer instruction. This approach is a direct attempt to imbue computers with the qualities of expert human tutors. Of all the approaches discussed here it is the one most directly relevant to filling the gap created by an insufficient supply of logs in instruction. This line of development grew out of early concern with just how long it took, and how expensive it was, to generate items for computer presentation. Early estimates of the amount of time required to produce one hour of computer instruction ranged from 77 to 714 hours on PLATO, 200-400 hours on TICCIT, and around 475 hours for the IBM 1500 Instructional System (Orlansky and String, 1979). One solution to this problem was sought by those who noticed that the process of preparing items for computer presentation was boring, repetitious, and dull--in other words a perfect job for computers. The resulting solution took the form of programs that would generate items for students (e.g. Koffman and Blount, 1974) and was called Generative Computer-Assisted Instruction although what we now mean by generative computer instruction is a little more sophisticated. In any event, it occurred to early observers of the scene that inasmuch as we were trying to use computers to mimic the item generation capabilities of expert human tutors, why not use computers to mimic all the capabilities of human tutors? Thus was born the notion of computerized tutorial dialogue.

In discussing tutorial dialogue approaches based on intelligent instructional systems, it is usually necessary to point out hastily that no distinction is intended between "intelligently" designed systems and "unintelligently" designed systems. Intelligent instructional systems may be as unintelligently designed as any others. Rather, the development of these systems is viewed as a specific attempt to apply artificial intelligence techniques to computer instruction in the sense of information structure oriented approaches discussed and advocated by Carbonell (1970) who contrasted these approaches with frame oriented approaches based on techniques of programmed instruction. Carbonell advocated an approach that would answer questions not specifically anticipated, construct appropriate questions on given topics, and carry on a "mixed-initiative" dialogue in which either the student or the computer could introduce a response, topic, or idea in a free and comfortable subset of English. This may sound like programming a computer to be an expert tutor, and it is meant to. The approach is in the mainstream of current developments in cognitive psychology which have taught us--or reminded us--that perception and learning are overwhelmingly constructive processes (cf. Resnick, 1983). In perception we do not collect bits of information from the "outside world" and paste them up on perceptual templates, and in instruction we are not writing information on blank slates in students' heads. Instead we are dealing with active and very rich simulations of the world which students must create in order to perceive or learn. It is analysis by synthesis with a vengence, and what gets transmitted in communication and instruction are not bits of information or knowledge but cues that may or may not be used to adjust the simulations being built up by students. The attempt in tutorial dialogue approaches is to deal directly with these simulations in ways that no drill and practice program--and no book--can.

Computers are both very good at this and very bad. Consider the following senten[illegible]

The [illegible] owned [illegible]t died.

This is a difficult [illegible] for us to parse. We quickly become entangled in its syntactic nestings. With typical human chauvinism many of us would assume that since this sentence is difficult for us to parse, it is impossible for a machine. Yet a computer could very quickly discern, after diving into its recursive routines for processing nested constructions, that there was a dog that was owned by a girl, that the dog bit a man, and that the man subsequently died.

Here is another example:

The man kicked the ball kicked the ball.

This is a perfectly grammatical sentence as any self-respecting machine would discover after reversing an English transformational rule for deleting function words and determining that a man to whom a ball was kicked, kicked the ball back. In both these examples, a computer is less likely than we are to be confused or distracted, and its ability to process these two examples illustrates real intellectual ability. "Artificial intelligence" is after all a poor name for the business of making computers intelligent. Intelligence, or intellectual ability, is really what the field is all about. That theories of intelligence are tested by algorithmization and putting them on computers is merely an issue of methodology, albeit a central one; there is nothing artificial about the capabilities targeted by this work.

Next we m[illegible] he following example, stolen from Donald Norman (197[illegible]):

What was Charles Dickens' telephone number?

A knowledgeable program would search the attributes it had associated with Charles Dickens looking for a telephone number, and/or it would search its list of telephone numbers looking for one associated with Charles Dickens. Presumably, no telephone number will be found for Charles Dickens, and this fact will simply be reported. As most human information processors must be aware, there is a more intelligent answer to this query that completely avoids searching lists of teleph[illegible] n[illegible]bers and/or attributes. To reply that telephones were not [illegible] Dickens' lifetime does require some knowledge, but this [illegible]ion could easily be stored by computer. The real problem is to devise a computer program that knows when this information is relevant.

These examples also illustrate the distinction drawn by John Brown, between "intelligent" computer instruction and what might be called "knowledgeable" computer instruction. We all seem susceptible to the confusion of knowledge and intelligence. We expect someone who is intelligent to know that relativity may have more to do with physics than with [illegible]mily ties, that battery may involve either an ele[illegible]trical or [illegible] legal charge, and that scales are relevant t[illegible] fish [illegible]. However, there are many intelligent people who h[illegible] no[illegible] of the th[illegible]y [illegible]lativity the legal aspects of fisticuffs, or B-flat minor, just as there are many educated people whose inferential capabilities are less than awesome. A computer program that could process the first two examples above would qualify as knowledgeable. However, only by correctly processing the last example would a computer qualify as intelligent.

In this light we might consider the following student/computer protocol taken from Brown's SOPHIE system (Brown, Burton, and Bell, 1974). In this example, the student is completing a series of tests on a computer simulated piece of electronic equipment (a regulated power supply) with a fault generated by the computer. Student responses are underlined.

<u>WHAT IS THE BSE EMITTER VOLTAGE OF THE VOLTAGE LIMITING TRANSITOR?</u>

BASE
TRANSISTOR
THE BASE EMITTER VOLTAGE OF Q5 IS -.037 MILLIVOLTS.

The spelling corrector has detected and corrected two misspellings. The system must also know that Q5 is the voltage limiting transistor.

<u>WHAT ABOUT FOR THE CURRENT LIMITING TRANSISTOR</u>

THE BASE EMITTER VOLTAGE OF Q6 IS .349 VOLTS.

The meaning of thi[illegible] question is completely dependent on the context established by the prior sentence. The system uses this context to determine that the student is interested in not only the voltage measurements but the voltage across a particular junction.

<u>WHAT SHOULD IT BE?</u>

IN A WORKING CIRCUIT THE BASE EMITTER VOLTAGE OF Q6 IS .593 VOLTS.

Again context determines the meaning of the question. Moreover the system must infer that "should" implies what the measurement would be in a properly working device.

<u>REPLACE Q6</u>

I AM GOING TO ASK YOU SOME QUESTIONS ABOUT HOW Q6 IS FAULTED. ARE ANY JUNCTIONS SHORTED?

At this point SOPHIE knows that the student is barking up the wrong tree. Rather than simply replacing the transistor as requested, SOPHIE begins [illegible] query the student about what he or she thinks is wrong. [illegible] student gets at least something right, the system sa[illegible] actually wrong and goes ahead and replaces the faulty pa[illegible]

Clearly, SOPHIE has progressed beyond knowledgeable syntactic and [illegible] student inquiries and is behaving in an "inte[illegible] system tries to create a reactive environment in which the s[illegible] actively trying out ideas in interaction with a simulation program. However, the program does more than simply simulate the information to be transmitted, it provides for tutorial feedback and, in effect, for a one-to-one relationship with an "articulate expert" problem solver who helps the student create, experiment with, and debug his or her own ideas.

Several reviews of this area have appeared, notable among which are discussions by Peelle and Riseman (1975), Bregar and Farley (1980), and Barr and Feigenbaum (1982). Carbonell's SCHOLAR (1970) and Brown's SOPHIE (Brown, Burton, and Bell, 1974) were of course seminal systems in this line of development.. The two premier systems currently seem to be GUIDON (Clancey, 1979) and Steamer (Stevens, Roberts, and Stead, 1983).

GUIDON serves as a physician's consultant for the student, who plays the role of the physician, in diagnosing infectious diseases. GUIDON focuses directly on the problems a subject matter expert faces in making his or her expertise and understanding accessible to students. GUIDON employs students' knowledge and interests in choosing what to present, it incorporates a knowledge base that is augmented to better organize and explain the subject matter to the student, and its teaching expertise is represented explicitly and modularly so that it can be modified (or tinkered with) for different research designs. GUIDON both "knows" the subject matter and can explain to the student the paths it uses to reach a diagnosis just as an expert tutor does.

Steamer is a computer-based system being developed by the Navy to provide instruction in steam propulsion engineering. It links a very complicated and highly abstract, quantitative (mathematical) model of a ship's steam propulsion system to high quality visual (graphics) representations of the underlying model. The student is thereby able to manipulate the underlying abstract model through the graphics interface and see the effects of these manipulations in visual form as they would be propagated throughout the steam propulsion system. However, and beyond this, Steamer uses the student's manipulations of the visual representation of the ship's steam propulsion system to better model his or her understanding of the physical system and to use this information to extend, correct, and deepen that understanding.

At this point, we may all wonder if we are going to see tutorial dialogue systems of this sort in our classrooms in the near future. About a year ago one of the major figures in the tutorial dialogue world passed through Oregon State University leaving the following quote in his wake:

> "It's amazing what you can do when you only have two megabytes of memory."

To those of us used to working with 32K and 64K byte personal computers, the notion of 128K bytes seems like Nirvana. Two million bytes is beyond all imagining, and this is apparently the low end for someone working with tutorial dialogues. The point to be made, of course, is that the computational requirements for tutorial dialogue systems are very large. A single us[illegible] system sufficiently powerful for delivery but not development of tutorial dialogues might be purchased today for about $20,000. Mark Hopkins might be cheaper. In ten years, of course, the picture will change completely, and for this reason the development of tutorial dialogue systems should be pursued vigorously now on large machines.

Somewhere among all the new directions for computer courseware a major breakthrough will occur. Tutorial dialogues appear to be a likely area in which to look for this breakthrough. This direction represents an approach that is both evolutionary and revolutionary. That is to say we can expect it to help us accomplish what we want to do now, but it may also alter in very fundamental ways our understanding of what instruction should be. In any event, tutorial dialogues could not be implemented without computers and they are in fact limited by the current state-of-the-art in both computer hardware and software. It is often said that hardware and software developments are far in advance of our capabilities to use them in instruction. In the case of tutorial dialogues, this is not true. We are both pushing and capitalizing on the state-of-the-art in computer hardware and software technology. Much still needs to be done: we need to learn how to represent imperfectly understood and poorly described knowledge domains, we need to reduce the costs in creating knowledge domains, better natural language processing must be developed, techniques for modeling learners must become far more sophisticated, our understanding of what master tutors and teachers do must be greatly enhanced, we need to learn how to interface computer tutorial dialogues with the practice of classroom teachers, etc. However, these issues only indicate that breakthroughs in this area will occur perhaps later rather than sooner. The promise of tutorial dialogues of this sort remains.

This promise is particularly evident when we review efforts to join tutorial dialogue techniques with simulation, the topic of the next section. In fact, we have already skirted these shoals fairly closely. After all, the student troubleshoots a simulated power supply in SOPHIE, diagnoses an ailing simulated patient in GUIDON, and operates a simulated steam propulsion system in Steamer. It may be past time to turn to the area of simulation in instruction.

Simulation

The currently strong and growing interest in simulation used for education is far overshadowed by the interest in and support for simulation used in training, more precisely military and industrial training. Most readers will be familiar with the long history and use of multi-million dollar aircraft simulators--some costing more than the aircraft they simulate--by the military and by aircraft manufacturers for pilot training. Twenty years ago if one mentioned the use of simulators in instruction the reference would have to be to aircraft simulators and nothing else. The advent of computer technology has permanently altered this state of affairs.

Because current simulators are based on programmable computers, they need not be single purpose, representing, for instance, only the cockpit of an F-14 fighter aircraft. Instead a wide range of related systems can be simulated for the purposes of training individuals who must learn to operate and maintain them. The Navy's Generalized Maintenance Trainer/Simulator (Rigney, Towne, King, and Moran, 1978) is a case in point. The GMTS can be used to simulate any device in which signal paths and their relationships to controls, indicators, and test points can be defined. So far the GMTS has demonstrated its versatility by being used to teach technicians to maintain both a radar repeater and a UHF communications system.

Again because current simulators are based on programmable computers, they can be much smaller and less expensive than they were originally. Simulators too are benefitting from the microelectronics revolution. The idea of "suitcase exercisers" abounds in today's military. MITIPAC (Rigney and Towne, 1977), for instance, took the GMTS and shrunk it down via micro-electronics to fit into a suitcase size package which provides a true job-site training capability. MITIPAC can now be transported to locations where military jobs are actually performed--in the field, on ships, on flight lines--and tailored to the specific jobs at hand.

Many simulators have been built, tried, and evaluated in training as Orlansky and String showed for training aircraft pilots (1977) and for training maintenance technicians (1981). In this sense, simulation is an established and proven technique for instruction. However, development of simulation for instruction is far from finished. The field is particularly fortunate in that promising and dramatic new "functionalities" now exist. New directions in computer instruction using simulation can be based on these functionalities. Three of these new functionalities are: interactive movies, surrogate travel, and spatial data management. All three of these use computer-controlled videodiscs.

Interactive Movies. Interactive movies attempt to translate movie viewing into an active, participatory process. In effect, the viewer becomes the director and controls many features of the movie. Feature controls available to the viewer are the following:

1. Perspecti . The movie can be seen from different directions. In effect, the viewer can "walk around" ongoing action in the movie or view it from above or below.

2. Detail. The viewer can "zoom in" to see selected, detailed aspects of the ongoing action or can "back off" to gain more perspective on the action and simultaneous activity elsewhere.

3. Level of Instruction. In some cases, the ongoing action may be too rich in detail or it may include too much irrelevant detail. The viewer can hear or see more or less about the ongoing process by so instructing an interactive movie system.

4. Level of Abstraction. In some instances the viewer may wish to see the process being described in an entirely different form. For example, the viewer might choose to see an animated line drawing of an engine's operation to get a clearer understanding of what is going on. In some cases, elements shown in the line drawings may be invisible in the ongoing action--for instance, electrons or force fields can be shown.

5. Speed. Viewers can, of course, see the ongoing action at a wide range of speed, including reverse action and no action (still frame).

6. Plot. Viewers can change the "plot" to see the results of different decisions made at selected times during the movie.

Surrogate Travel. Surrogate travel forms a new approach to locale familiarization and low cost instruction. In surrogate travel, images organized into video segments showing discontinuous motion along a large number of paths in an area are stored on videodisc. Under microprocessor control, the student accesses different sections of the videodisc, simulating movement over the selected path.

The student sees with photographic realism the area of interest, for instance a city street or a hallway in a building. The student can then choose both the path and the speed of advance through the area using simple controls, usually a single joystick. To go forward the student pushes forward on the joystick, to make a left turn the student pushes the joystick to the left; to go faster the student pushes the joystick harder; etc.

The videodisc frames the viewer sees originate as filmed views of what one would actually see in the area. To allow coverage of very large areas, the frames are taken at periodic intervals that may range from every foot inside a building, to every ten feet down a city street, to hundreds of feet in a large open area (e.g., a harbor). Coverage of very small areas is also of interest. In microtravel, which is a combination of surrogate travel and interactive movies, travel is possible where humans could never go--inside watches while they are running, inside living organisms, etc.

The rate of frame playback, which is the number of times each video frame is displayed before the next frame is shown, determines the apparent speed of travel. Free choice in what routes may be taken is obtained by filming all possible paths in the area as well as all possible turns through all intersections. To some extent this is a time consuming and expensive technology, but it has become relatively efficient because of the design of special equipment and procedures for doing the filming.

Demonstrations of this technology have been developed for building interiors (National Gallery of Art), a small town (Aspen, Colorado), an industrial facility (nuclear power plant), and San Francisco Harbor. Plans are underway to produce a prototype video map library of broader scope for selected areas worldwide.

Spatial Data Management. Basically, spatial data storage and retrieval of information is the method of loci transformed to a video, or computer graphics, format. The information is stored and retrieved through its association with already familiar geographic terrain.

Suppose, for instance, a student wanted to study the musical environment in which Ralph Vaughn Williams wrote his "Concerto for Tuba and Orchestra." In an ordinary data retrieval system the student will type in a complicated set of Boolean expressions--or English phrases standing for Boolean expressions--and will receive in return only textual information about the topic. Relevant information closely related to the information successfully retrieved will not appear unless the student starts from the top again with a new set of Boolean expressions. In a spatially organized data system, the underlying geography will be familiar to the student, for instance the school campus. The student may then "fly" to the music department (or library, concert hall, professor's office, etc.) and look for a tuba (or an orchestra, music library, portrait of the composer, etc.). Upon finding a tuba or other relevant cue, the student can "zoom" into it, still using his single joystick control, select the concerto by name (or by hearing it, seeing the score, seeing the composer, etc.) and then hear, see, and read more information about it all retrieved through visually oriented associations.

In this way, spatial data management acts as an electronic library that provides students and instructors access to a wide assortment of multi-source and multi-media information whose components are associated in a natural and easily accessible manner. Instructors can access the system to create and/or assemble their own information spaces to be explored later by their students or subsequently present these materials to large audiences in single locations using large screen television projection or to multiple locations through cable distribution systems. Students can independently use the system for individualized instruction by working through previously designed information spaces, by browsing on their own, or by creating their own data spaces. When students and instructors are in remote locations, offsite instruction can be facilitated by linking two or more systems together using regular telephone lines. In this manner, a student or instructor can "fly" the other to a topic of interest, sharing at geographically remote sites a large, visually oriented library of information.

Two points are worth noting about these new directions for simulation applied to instruction. First, they cannot be implemented in a book--or on a log. Implementing surrogate travel experiences on a log, for instance, would require equipping it with wheels, wings, and a rudder. This is to say nothing of the log's increase in operating costs, which would be considerable. Second, the application of these new directions for simulation-based computer instruction in education is just beginning. One can easily imagine application of this technology to science education and perhaps a few words on this application are in order.

The best way to learn science is by doing it. The excitement, mystery, frustrations, and triumphs of science are only dimly revealed by the usual fare of introductory science courses. It would be far better for students, especially introductory students, to approach science with freedom to indulge their curiosity, form and re-form their own hypotheses, design and perform their own experiments, and build their own models and theories to explain natural phenomena. Unless there are drastic shifts in national funding policies for science education, this essential scientific experience will be prohibitively expensive to provide. The result is that students--especially elementary and junior high school students--are "turned off" by science at a time when our industrial and academic need for scientists, engineers, and technologists is both great and increasing.

What is needed in science education is something that has the impact of video gaming, but at the same time possesses substantial pedagogical power. One way to accomplish this is to provide simulated scientific experiences to students. Good simulations are exciting, compelling, and teach effectively by providing an environment in which learners must live with their decisions. Simulated experiences need not replace existing laboratory and field exercises, but they may expand and supplement them. Moreover, in at least four ways simulated experiences may be superior to real experiences. First, and primarily, simulation can be economical. Use of simulation should reduce the need for laboratory equipment and its maintenance, laboratory supplies, and travel costs for field experience. Second, simulation can make relevant phenomena more readily visible in two ways. In one way it can make the invisible visible. For instance, the flow of ions can be seen more clearly and simply under simulated conditions than under real conditions. In another way, simulation may increase the visibility of a phenomenon by separating it from a confusing and chaotic background. One can see the conceptual forest without getting lost in the procedural trees. Third, simulation allows reproducibility. Students can replay chains of events over and over that they could not otherwise observe repeatedly. Fourth, simulated experience is often safer than the real thing. Airplanes can be crashed, poisons can be ingested, and laboratories can be exploded with impunity in simulated environments. Such behaviors in real environments are obviously impractical.

Two sorts of relevant scientific experience that lend themselves readily to simulation are field study and laboratory experimentation. These two kinds of experience could be provided using the new functionalities described above. These functionalities could be used to "build" video field trips and simulated laboratories.

In the field, the student sees the total ecological view. He/she sees the overall landscape, the terrain, the populations of organisms, and individual samples of interest in their special areas. In sciences such as biology, geology, paleontology, archaeology, and even astronomy, substantial learning and appreciation can be achieved by travel to locations that are difficult to access under the best of conditions. However, field trips are treated as an instructional frill. After all, the trips are made rarely and locally--they depend for success on what is serendipitously near by--; they emphasize only the group--individuals do not have an opportunity to "do" the science on their own--; and most of the administrative effort centers on getting to the field and getting back, not on the field experience itself. As a result, even short, local field trips are being cancelled by schools because their cost in time and fuel is not balanced out by their educational return. Surrogate travel removes the major objections to field experience and offers to each student broadened opportunities to experience scientific phenomena in their natural, ecological context.

Students interested, say, in the biology of deserts could visit the Gobi in the morning, the Sahara around noon, and the Sonoran in the afternoon. They could travel around in each habitat locating, identifying, and "gathering" samples roughly in the same way, and for the same purposes, as a trained scientist. Panning and zooming through the full range of habitats could develop in students many of the same intuitions and understandings of environmental, geographic, climatic, etc., contexts that an experienced scientist gains from actual travel.

Back in school, laboratories provide a problem solving environment where students interact, observe processes, and are stimulated to synthesize concepts as part of their learning. However, many schools are eliminating laboratories from their science courses, not because they are not useful learning experiences, but because of the cost of obtaining, maintaining, and supporting specimens and samples and because of the cost and effort involved in purchasing and maintaining laboratory equipment. Interactive movies and spatial data management allow us to simulate laboratory experiences without the high cost and effort that is normally involved under the present pattern.

Students could create, store, and retrieve information from mammoth data banks using spatial data management. One can imagine high school students organizing an entire archaeological excavation or geological survey using spatial data techniques. One can also imagine elementary school students setting up and running high-energy particle physics experiments through interactive movies with plot control. Students would also have full use of the latest in telescopes, microscopes, and even endoscopes through computer-based simulation.

Finally laboratory and field experiences could be linked so that hypotheses developed in the laboratory would be tested by return "travel" to the correct habitat, "collection" of the proper (or, even improper) data or specimens, and return to the laboratory for testing and verification. In this way, the excitement, frustrations, and triumphs of scientific experiences would become accessible to students.

In the above, simulation was presented as a new direction that is finding its way into computer instruction, but it is interesting to note that the history of computer instruction is exactly the reverse. The first use of computers to teach grew out of a computer-based system that was primarily intended for simulation of real world experiences. This was the Air Force's SAGE (Semi-Automatic Ground Environment) system which was built in the late 1950's to train Air Force personnel in the techniques and tactics of air defense (Rowell and Streich, 1964; Parsons, 1972). Computers in SAGE were initially used to stimulate equipment, mostly radar, to which ground-based air defense personnel were to make appropriate reactions. However, as time progressed the SAGE computers began to be used to present training in a more general-purpose fashion.

The University of Illinois PLATO (Programmed Logic for Automatic Teaching Operations) was probably the first computer system built specifically for computer instruction. Interestingly, it too was first supported solely by the military--in this case by the Army Signal Corps, the Office of Naval Research, and the Air Force Office of Scientific Research (Bitzer, Braunfeld, and Lichtenberger, 1962). Initially PLATO was used as a sort of "book with feedback" following the suggestion of Chalmers Sherwin, but it was early infested with fruit flys. Few who saw early demonstrations of PLATO III in the late 1960's were able to escape the fruit fly demonstration. This was a simulated biology laboratory showing in high quality graphics successive generations of fruit flys as they revealed visually the workings of an underlying model of genetics. This early use of simulation in computer instruction, of course, continues to the present.

The focus in this section is on new techniques for simulation, three of which are listed above. These three have been discussed in a little more detail by Bolt (1979) and by Levin and Fletcher (1981). Other techniques may well be on the way. We have barely begun to explore the instructional possibilities of natural language processing (as opposed to computer language processing), voice output, voice input, computer-generated imagery (which may obviate some of the need for videodisc storage), and psychoneurological monitoring. New "functionalities" for these capabilities will doubtless be developed. However, it should be emphasized because it is so frequently overlooked that this process of discovery is at least as demanding of time, resources, and ingenuity as the development of the computational capabilities themselves. Swamping schools with hardware and computer capabilities and then expecting instructional functionalities to flow spontaneously in their wake is simply wrong. This is a start, but only that. However, the other side of the coin also needs considering and this is done in the next, and final, section.

Final Word

In 1960 T. F. Gilbert wrote:

> If you don't have a gadget called a "teaching machine," don't get one. Don't buy one; don't borrow one; don't steal one. If you have such a gadget, get rid of it. Don't give it away, for someone else might use it.
>
> This is the most practical rule, based on empirical facts from considerable observation. If you begin with a device of any kind, you will try to develop the teaching program to fit that device" (P. 478, the italics are Gilbert's).

This is a point of view with which many of us will have substantial sympathy. Educators who have mastered their craft through considerable investment of time and energy in learning how to use the traditional technologies of text, lectures, blackboards, and real-equipment laboratories have every right to be suspicious of new technology that threatens to revolutionize the hard-won techniques now at hand. Even programmers, initiates into the priesthood of computer technology, have learned to be wary of devices that turn normally sane and calm individuals into maniacs who, because the machines do what they are told to do and not what they are wanted to do, are willing--and eager--to destroy thousands of dollars of equipment with their bare hands. Moreover Gilbert is undoubtedly correct when he suggests that we may develop teaching programs to fit the technology at hand. Of course we will, and, to some extent, we always have. To suggest that we should not pursue new technologies for this reason may not be so correct.

In other words, it may be as wrong to inundate our educational institutions with new technologies without insisting that they do at least something to help us through the day as it is to hold off all investment in new technologies because they may affect what it is we want to do. The correct approach seems to be somewhere in the middle. No one envisioned teleconferencing when the telephone was invented, no one imagined our current interstate highway transportation system when the horseless carriage came along, and steam engines languished for 30 years pumping water out of coal mines before someone began to think seriously of their possibilities for self-locomotion. We have benefited from the introduction of these devices into our lives just as we have suffered from them. We must give the new technologies their place if we are to improve our instructional practice as the Gardner Commission said we must ("A Nation At Risk," 1983), but, at least in the case of computers, we are in a position to insist that they be of some immediate practical value along the way. This is a fortunate position to be in and we should capitalize on it. Computers can meet the goals and solve some of the current problems of schools and school districts today at the same time that they are helping us advance our craft of instruction. We can and should expect them to do both.

Finally, computers will help us compensate for an insufficient supply of logs in instruction--they will help us better perform the business of instruction as we envision it today. They will also broaden our horizons. They will change and expand our ideas about what instruction is and what it must do. Their challenge to us as educators is as serious as their promise. We should rise to the occasion.

References

Atkinson, R. C. and Paulson, J. A. An approach to the psychology of instruction, Psychological Bulletin, 1972, 78, 49-61.

Barr, A. and Feigenbaum, E. A. Handbook of Artificial Intelligence, Volume 2. Los Altos, CA: William Kaufman, 1982.

Becker, H. J. Microcomputers in the Classroom: Dreams and Realities. Eugene, OR: International Council for Computers in Education, 1983.

Bitzer, D. L., Braunfeld, P. G., and Lichtenberger, W. W. PLATO II: A multiple-student, computer-controlled, automatic teaching device. In, J. E. Coulson (Ed.), Programmed Learning and Computer-based Instruction. New York, NY: John Wiley and Sons, 1962.

Bolt, R. A. Spatial Data Management. Cambridge, MA: Massachusetts Institute of Technology, 1979.

Bregar, W. S. and Farley, A. M. Artificial intelligence approaches to computer based instruction. Journal of Computer Based Instruction, 1980, 6, 106-114.

Brown, J. S., Burton, R. R. and Bell, A. B. SOPHIE: A sophisticated instructional environment for teaching electronic troubleshooting (An example of AI in CAI) (BBN Report 2790). Cambridge, MA: Bolt Beranek and Newman, Inc., 1974.

Bunderson, C. V. Courseware. In H. F. O'Neil, Jr. (Ed.) Computer-based Instruction: A State-of-the-art Assessment. New York, NY: Academic Press, 1981.

Bunderson, C. V., Baillo, B., Olsen, J. B., Lipson, J. I., and Fisher, K. M. Instructional effectiveness of an intelligent videodisc in biology. Machine-mediated Learning, in press.

Carbonell, J. R. AI in CAI: An artificial intelligence approach to computer-assisted instruction. IEEE Transactions on Man-Machine Systems, 1970, 11, 190-202.

Chant, V. G., and Atkinson, R. C. Optimal allocation of instructional effort to interrelated learning strands. Journal of Mathematical Psychology, 1973, 10, 1-25.

Clancey, W. J. Tutoring rules for guiding a case method dialogue. International Journal of Man Machine Studies, 1979, 11, 25-49.

Crowder, N. A. Automated tutoring by means of intrinsic programming. In E. H. Galanter (Ed.), Automatic Teaching: The State of the Art. New York: John Wiley and Sons, 1959.

Dearman, N. B. and Plisko, V. W. The Condition of Education. Washington D.C.: U.S. Government Printing Office, 1980.

Fletcher, J. D. Models of the learner in computer-assisted instruction. Journal of Computer-Based Instruction, 1975, 1, 118-126.

Fletcher, J. D. Computer-assisted instruction in beginning reading: The Stanford projects. In L. B. Resnick and P. A. Weaver (Eds.), Theory and Practice of Early Reading, Volume 2. Hillsdale, NJ: Lawrence Erlbaum Associates, 1979.

Fletcher, J. D. and Beard, M. H. Computer-assisted instruction in language arts for hearing-impaired students (Technical Report No. 215). Stanford, CA: Institute for Mathematical Studies in the Social Sciences, Stanford University, 1973.

Gilbert, T. F. On the relevance of laboratory investigation of learning to self-instructional programming. In A. A. Lumsdaine and R. Glaser (Eds.), Teaching Machines and Programmed Learning: A Source Book. Washington, DC: National Education Association, 1960.

Koffman, E. B. and Blount, S. E. A modular system for generative CAI in machine-language programming. IEEE Transactions on Systems, Man, and Cybernetics, 1974, 4, 335-343.

Laubsch, J. H. Optimal item allocation in computer-assisted instruction. IAG Journal, 1970, 3, 295-311.

Levin, S. and Fletcher, J. D. Videodisc technology: An approach to the design of training devices. In J. Moraal and K. F. Kraiss (Eds.), Manned Systems Design: Methods, Equipment, and Applications. New York, NY: Plenum Press, 1981.

Lorton, P. V. Computer-based instruction in spelling: An investigation of optimal strategies for presenting instructional material (Doctoral Dissertation). Stanford, CA: Stanford University, 1973.

A Nation at Risk: The Imperative for Educational Reform. Washington, DC: U.S. Government Printing Office, 1983.

Norman, D. A. Memory, knowledge, and the answering of questions. In R. L. Solso (Ed.), Contemporary Issues in Cognitive Psychology: The Loyola Syposium. Washington, DC: V. H. Winston and Sons, 1973.

Orlansky, J. and String, J. Cost-effectiveness of flight simulators for military training (IDA Paper P-1275). Arlington, VA: Institute for Defense Analyses, 1977.

Orlansky, J. and String, J. Cost-effectiveness of computer-based instruction in military training (IDA Paper P-1375). Arlington, VA: Institute for Defense Analyses, 1979.

Orlansky, J. and String, J. Cost-effectiveness of maintenance simulators for military training (IDA Paper P-1568). Arlington, VA: Institute for Defense Analyses, 1981.

Papert, S. Mindstorms. New York, NY: Basic Books, 1980.

Parsons, H. M. Man-machine System Experiments. Baltimore, MD: Johns Hopkins Press, 1972.

Paulson, J. A. An evaluation of instructional strategies in a simple learning situation (Doctoral Dissertation and Technical Report No. 209). Stanford, CA: Institute for Mathematical Studies in the Social Sciences, Stanford University, 1973.

Peelle, H. A. and Riseman, E. M. Four faces of HAL: A framework for using artificial intelligence techniques in computer-assisted instruction. IEEE Transactions on Systems, Man, and Cybernetics, 1975, 5, 375-380.

Resnick, L. B. Mathematics and science learning: A new conception. Science, 1983, 220, 477-478.

Rigney, J. W. and Towne, D. M. Some concepts, devices and procedures for improving technical training and on-the-job performance of maintenance (Technical Report 76-C-0023-1). Orlando, FL: Naval Training Equipment Center, 1977.

Rigney, J. W., and Towne, D. M., King, C. A., and Moran, P. J. Field evaluation of the generalized maintenance trainer-simulator: I. Fleet communications system (Technical Report No. 89). Los Angeles, CA: Behavioral Technology Laboratories, University of Southern California, 1978.

Rowell, J. T. and Streich, E. R. The SAGE system training program for the Air Defense Command. Human Factors, 1964, 6, 537-548.

Rumelhart, D. E. The effects of interpresentation intervals on performance in a continuous paired-associate task (Technical Report No. 116). Stanford, CA: Institute for Mathematical Studies in the Social Sciences, Stanford University, 1967.

Stevens, A., Roberts, B., and Stead, L. The use of a sophisticated graphics interface in computer-assisted instruction. IEEE Computer Graphics and Applications, 1983, 3, 25-31

Suppes, P. Problems of optimization in learning a list of simple items. In M. W. Shelly and G. L. Bryan (Eds.), Human Judgments and Optimality. New York, NY: John Wiley and Sons, 1964.

Suppes, P. Some theoretical models for mathematics learning. Journal of Research and Development in Education, 1967, 1, 5-22.

Suppes, P., Fletcher, J. D., and Zanotti, M. Performance models of American Indian students on computer-assisted instruction in elementary mathematics. Instructional Science, 1975, 4, 303-313, b.

Suppes, P., Fletcher, J. D., and Zanotti, M. Models of individual trajectories in computer-assisted instruction for deaf students. Journal of Educational Psychology, 1976, 68, 117-127.

Suppes, P. and Morningstar, M. Computer-assisted instruction. Science, 1969, 166, 343-350.

Williams, M., Hollan, J., and Stevens, A. An overview of STEAMER: An advanced computer-assisted instruction system for propulsion engineering. Behavior Research Methods and Instrumentation, 1981, 13, 85-90.

TEACHER EDUCATION FOR COMPUTERS IN THE SECONDARY SCHOOL CURRICULUM

(A panel discussion)

with
David Moursund
Doris Carey
Regan Carey
Keith Wetzel

Introduction

by David Moursund

There are many factors that relate to a school's progress in making effective instructional use of computers. Perhaps the four most frequently cited factors are the nature and extent of available:

1. hardware
2. software
3. curriculum materials
4. teacher knowledge and skills.

All of these factors are important, and one can cite others. Moreover, it is becoming increasingly clear that a school must make appropriately balanced progress in all these areas if computers are to have a significant impact upon students and their curriculum.

Colleges of education, centers for continuing education, school districts and private industry are addressing teacher education issues. The University of Oregon and the school districts serving Eugene, Oregon provide good examples of what is being done and/or can be done. There has been substantial cooperation among these organizations in attacking the preservice and inservice teacher education problem.

Of considerable interest is the University of Oregon's long term involvement with these problems. The UO started its Master's Degree program in computer science education in 1970. The first candidate to its doctorate program in computers-in-education was admitted in 1971. In recent years the College of Education and the Department of Computer & Information Science have offered a considerable variety of courses at the undergraduate and graduate levels. Close cooperation has been developed between these two units of the university. This has led to computers-in-education coursework that reflects the strengths of both of these educational bodies.

The College of Education has carried out substantial planning for computers-in-education. Four general goals have emerged and substantial progress is occurring toward their accomplishment. These are:

1. Require all students in the undergraduate teacher education program to develop a functional level of computers-in-education literacy.

2. Provide appropriate strands of study so that students in the teacher education program can take substantially more computer related coursework (beyond requirement 1. above), if they so desire.

3. Provide solid graduate level computer related courses for both the practitioner and potential researcher in each division of the

College of Education.

4. Offer a substantial program of study leading to the Master's degree and the Doctorate in computers-in-education.

It is hoped that this conference and this panel will help other colleges and universities as they work to develop their own programs of computer related study for preservice and inservice education.

It is the purpose of this panel to focus on teacher education. Therefore, let us look a few years into the future and assume that significant progress has occurred for each of the first three factors mentioned in the opening paragraph. For example, let us assume that a typical secondary student has an average of 1 - 2 hours of computer access per week. Let's assume that the very best of current (1983) software is readily available, along with some software that is just coming off the drawing board. Let's assume that curriculum materials, including textbooks from all disciplines, are beginning to reflect computers as an aid to problem solving, as a source of potential problems, and as a topic of study.

One last assumption, before we raise the teacher education issue. Let us assume that the typical secondary school student has had substantial exposure to computers while in grades K-6. Moreover, this student has taken a half year long computer course while in grades 7 or 8. Now, what do the secondary school teachers need to know about computers?

This question is relatively easily answered for the teacher of computer science and/or computer programming. Here we have many years of college level experience to draw upon. Moreover, computer science and computer programming have been taught in quite a few secondary schools for many years.

It is in other disciplines, such as science, mathematics, social science, special education, and so on where the question becomes more difficult to handle. The other members of this panel will address these areas.

Computer-Oriented Inservice and Preservice Training for Mathematics Teachers and Science Teachers

by

Doris Carey

In the late seventies, the National Council of Teachers of Mathematics declared that its goal for the eighties was to shift the emphasis of mathematics instruction away from rote learning in favor of problem solving. The pursuit of this goal implies dramatic changes for teacher education. Rapid advances in technology and the public demand for improved instruction in mathematics and science are also bound to cause changes in the skills and knowledge that we expect of children. But these changes do not merely imply a shift in curriculum. They will require a major restructuring of teacher education programs. Teachers will need new skills and assume new roles. Institutions of teacher education will themselves require great changes of emphasis, new tools, and new roles for the leaders in the field of teacher education. The question is no longer whether change is needed, but how quickly and efficiently we can prepare teachers for change.

The traditional classroom has long been a miniature version of a lecture theatre. Indeed, the teaching styles of teachers have favored recitation, and there have been few movements in educational reform with impact sufficient to change the role of the teacher as lecturer. Research has shown that a full 80% of teacher questions deal with factual knowledge or classroom and behavior management. The focus has been on the teacher as the source of information.

The new technology is bringing new physical devices to the classroom, but the changes that are needed go far beyond the acquisition of a new tool or instrument. If tne computer becomes an information retrieval device, then it will augment and may even replace tne teacher as the source of information. The teacher's role is not then one of obsolescence, but of manager of the resource. Children will not merely ingest information at a greater rate. They will be required to assemble and sort facts, analyze them, and apply them to the solutions of problems. The teacher will need the skills to manage this facility and organize learning to maximize efficient, meaningful learning.

It is unrealistic to expect teachers to do their own learning in a traditional setting, then go forth and design learning experiences for children that are foreign to anything they have experienced themselves as learners. Unless teacher education changes from a lecture hall to a laboratory, then teachers will continue to model instruction on traditional, familiar settings.

There are many skills that must be mastered by the manager of the classroom laboratory. Teacher education institutions must continue to emphasize theories of group learning, individualized instruction, experiential learning, behavior management, and the expanding knowledge about these theories which is gained from research. The accumulated knowledge and experiences of the experts must be assembled to aid in the design of the learning environments of the future.

In addition to a suitable learning environment, children will need solid programs and realistic challenges. The shift to problem solving in the mathematics and science classroom will not occur simply as a result of making volumes of facts available to children. The curriculum changes must take place first in the minds of educators.

For the mathematics teacher, the change from rote learning to attainment of concepts is complex. The product of two numbers, for example, can easily be obtained in the classroom. What is more difficult, though, is the mastery of the concept of multiplication, and the planning of an environment where the concept can be applied meaningfully. Teachers will require a different set of strategies if they are to help students integrate facts with concepts. They will require a repertoire of techniques that inspire higher levels of thinking in their students.

Teachers of science will have rich possibilities for the simulation of problem-solving situations. In addition to the learning tools available today, they will see the microcomputer used to model problems such as air pollution and nuclear explosions, or to simulate structures such as molecules or sound waves. The computer will provide dynamic models which will supplement traditional laboratory equipment. Again, teachers will require skills of selection and integration of these new tools for meaningful learning environments for their students. Teacher education must include a variety of these technological tools, both hardware and software, as well as opportunities to practice the skills needed for efficient use and selection.

A learning medium that has become popular as a simulated problem solving environment is computer programming. There is much to learn about the transfer of skills to other areas and the thinking processes involved in the writing of a program, but it is generally felt that children of all ages develop certain mental functions when they learn to program. If time and research should succeed in showing how and why these facts are true, then it may well be the province of mathematics teachers to explore this field further in terms of applications to problem solving.

For example, the computer language LOGO has become increasingly popular in the elementary school classroom. In the majority of cases, the subset of the language which offers turtle graphics is receiving much of the attention. There is a general feeling that the learning of geometry is enhanced by discovering facts and concepts by experimentation, and the onerous label "mathematics" is slowly changing its reputation. Nevertheless, there are complex philosophical issues to be considered. There is no evidence that turtle graphics covers all aspects of mathematics, particularly if the discovery is to be left to student control. The implications for teachers of mathematics at both elementary and secondary levels are interesting and difficult. On one hand, we have a suitable "microworld" for exploration and discovery and, on the other, some desire for structure and standardization. Our system of education is based on our capacity for diagnosis and measurement, but these are rendered almost impossible, or at least impractical, in the discovery environment. Present and future teachers must face these issues, explore the possible solutions, and acquire the knowledge they need to question and make decisions about the issues.

As the technology changes, so do the issues and implications for educators. As public and political attention are focussed upon mathematics and science education, there is some hope that the new technology will become highly visible in these disciplines. It is time for those responsible for preservice and inservice to make a commitment to the redefintion of the roles of teachers and the goals of teacher education. Higher order thinking and strategies must be ready to challenge our teachers as well as our students.

New Tech in the Social Science and Humanities Classroom

by Regan Carey

The study of society in the current educational system cannot ignore the changes that new information handling technology, such as microcomputers and videodiscs, bring to our society. More importantly, the classroom can participate in these changes. This new technology has the potential to enhance some of the more traditional methods of humanities instruction, and to significantly alter others. However, offering effective instruction which utilizes the power of microcomputers is not a trivial task. It is unrealistic to leave educators to explore, discover, and develop new techniques without help and direction. It is largely the role of inservice and preservice training to provide that help.

A considerable portion of the instruction in the social studies and humanities classroom has traditionally dealt with children acquiring and practicing skills in the location and handling of information. Library skills and basic search strategies using indexes, bibliographies, encyclopaedias, and the like are usually considered to be fundamental.

The amount of accessible information available in our society is increasing daily, especially in the area of electronic information storage and transfer. Commercial on-line databases, such as The Source, Prestel, or CompuServe, are available to the general public at ever decreasing prices. Specialized databases for use on microcomputers, ranging from the Christian Bible to census statistics, are available. Retrieval of information from these newer forms of information storage is a skill that is increasingly valuable to a member of the information society. It falls within the responsibility of the educational establishment to provide training in the newer and more powerful areas of information retrieval. Leaving such instruction to the commercial market would be akin to leaving reading instruction to bookstores.

It is obvious that a teacher must be familiar with these skills before s/he can educate children in their use. It would be reasonable to assume that those responsible for inservice and preservice training must provide guidance and instruction for educators in the new skills. A likely entry point for educators would be instruction in the use of these powerful information handling tools as aids in their own tasks as teachers.

The time and effort demanded by clerical imperatives has dominated much of the instruction in the traditional classroom. In writing and composition using paper and pen, it is a rare language arts teacher who insists on more than two rough drafts from a student. The energy and commitment involved in multiple refinement of a piece of writing is simply too great. Clerical constraints dictate a hidden curriculum that encourages a child to be satisfied with a product that is often less polished than it could be.

Using word processing technology, error correction becomes a reasonably simple task. Both spelling and grammar can be refined to a degree that was impractical when confined to paper, pencil, and eraser. Another powerful facet of a word processor is its capability to allow the manipulation of entire chunks of a piece of writing. This electronic version of "cut and paste" can encourage the refinement process if presented to the student properly. By shouldering a significant part of the onerous clerical tasks involved in writing, the word processor permits the teacher to subject the product of a student's creativity to as many refinements as are considered necessary, without the implied penalty of "rewriting" an entire exercise. The hidden curriculum is bypassed, and an entirely new perspective of the refinement process can be promoted. The freedom permitted by word processing can result in a new attitude towards the redrafting of a piece of writing. Once it is a quick and trivial matter to subject a composition to refinement, a child can be taught to pursue excellence more readily.

The training of a child in the use of a word processor opens numerous new challenges to the educator. However, the keyboarding skills necessary for students to efficiently use the new technology must be stressed. Other forms of data entry may be forthcoming (e.g. voice entry), but they are unlikely to have an impact on the classroom use of the microcomputer in the near future. Questions arise when addressing the task of instruction in keyboarding skills. Is it the role of the "typing" teacher? If so, typing classes must be offered in the elementary grades, and be compulsory. This is not an overly practical approach. It falls then to the elementary teacher to introduce children to skill development in keyboarding. If young students are to be given access to keyboards, good keyboarding habits must be instilled at the same time. Ignoring this problem will not make it go away. It will simply create the major task of correcting bad habits further along in the educational process. Further research into superior techniques of instruction in these areas is constantly needed. Further training must be supplied to educators to help them in these new and relatively unexplored tasks.

Simulations and role play have long proved themselves as effective media for learning. Unfortunately, a simulation which accurately portrays reality must account for an enormous number of parameters. The traditional compromise in the social sciences classroom has been to limit the variables in a simulation to a "managable" number, at the expense of accurate simulation. Once again, the clerical aspects of a classroom activity have dictated its form. Once again, the microcomputer can be used to handle the greater proportion of the clerical duties, thereby freeing both children and teacher to concentrate on the business of learning from a simulation. This opens the door for simulations which present a more accurate model of the world, yet require less organizational time and effort to carry out. The creation of a suitable classroom environment for this type of advanced simulation is a major challenge to the classroom teacher. Social science instruction can become an interactive laboratory experience, with the potential for far greater conceptual understanding of many facets of society. However, any science teacher can attest to the fact that setting up and running an effective laboratory exercise is not a trivial task. Indeed, if there is inadequate preparation, the result is often a waste of time for both student aad teacher. Training in the art of setting up this type of learning environment is rarely given to the social science teacher. Yet, in the light of these new and powerful tools, such training is imperative. It is unrealistic to expect an educator to gain these skills through experience or through some form of osmosis. Those responsible for the design of preservice and inservice training must recognise the need and take steps to meet it.

The potential impact of the new technology on education is enormous. Relieved of much of the clerical tedium of today's educational process, both teacher and student are able to expand in directions that were not possible before. The ready availability of fast retrieval of information will soften the boundaries between separate disciplines, and require educators to reevaluate the role of memorized content in all curricula. An emphasis on problem solving and conceptual thinking should replace an emphasis on content. Much of modern social science instruction can become experimental and exploratory, without many of the constraints that limited previous attempts in that direction. The options are many and the consequences of action, or inaction, are great. Whatever decisions are reached by the educational system, adequate preparation and guidance should be made available to the teachers of the information age. It is the role of preservice and inservice training to provide it.

Training for Special Educators

Keith Wetzel

You may recall the story of Rip Van Winkle. There is a newer version. Rip Van Winkle came back after years of sleep and visited a supermarket and shopping mall. He saw many things that scared him--long aisles of frozen goods, bar code readers at the checkout counter, and escalators were among them. What was he to do? He decided to go to school, because in school things never change.

For special education however, things have changed. The advent of Public Law 94-142 has introduced key elements of education for handicapped students which have caused great change and have placed demands on school systems.

Special education teachers and administrators are faced with the management of a wealth of information. A major component of this law is an individualized educational plan (IEP) for each student. An IEP entails many processes and each process is usually represented by a form. They include:

-parent permission to test,

-notification of meeting to review test results,

-notification of meeting to plan an individualized program,

-numerous test results,

-goals, objectives, and criteria for evaluation.

What does this have to do with computers and the type of

ED 239 397 THE COMPUTER: EXTENSION OF THE HUMAN MIND II. PROCEEDINGS OF THE ANNUAL SUMMER CONFERENCE (EUGENE, OREGON, JULY 20-22, 1983).

3 OF 3 OREGON UNIV., EUGENE. COLL. OF EDUCATION.

83 239P.

training that should be offered teachers in both preservice and inservice? Computers do a very good job of processing information. Many of the processes involved in the IEP and the paper work they represent can be computerized. In a recent review of the IEP literature it was found that teachers can save time by using computers for

-maintaining student files,

-matching objectives to test results,

-graphing progress reports and pre/post test comparisions,

-storing scheduling information,

-guaranteeing confidentiality,

-writing reports using word processing capability.

The computerization of IEP's is possible, and it has been successfully accomplished by some school districts. What must be included in training to enable educators to use computers for these purposes? First, teachers and administrators must receive training in 'user' languages to be able to adapt presently available programs to their needs. One possibility would be to encourage educators to use electronic spread sheets and data based management systems to meet IEP management needs. The goal of this training is to enable educators to make software which is available commercially act as an IEP manager.

A second tactic would be to make the need known to commercial software developers and have a product designed to meet the needs of special education. However, the program would have to be operated in much the same way as the management programs mentioned previously, therefore, the same type of training would be necessary.

How much time is needed to train potential teachers to use information management systems? At the preservice level students interested in special education should have one four credit hour course in Computers in Special Education. Thirty percent of the of the course should be devoted to mastery of user languages for information management systems. For teachers already in the field, training would be tied to implementation of the specific management system adopted by the school district. The basic skills needed to operate the goals and objectives match component of an IEP program could be learned in a one day in-service workshop.

Usually, we think of IEP's in connection with the mildly handicapped; however, those who are working with the moderately and severely handicapped in sheltered workshops could use similiar management systems to meet their needs.

Although the focus of this panel is on secondary education, it is clear that IEP management is equally relevant to all levels of special education.

The second area to consider is individualized instruction. The teacher training to meet needs in individualized instruction is different from that needed for management systems. This area is very exciting because potentially it can combine the management of instruction with a tutoring environment made possible by expert system technology.

The advantages offered by the computer management of instruction have been demonstrated by math instruction systems. A student works on a lesson at the computer. Later, the teacher can receive a printout of the specific skills being taught by that lesson, the percentage answered correctly by the student, the time the student took to complete the lesson, and the place the next lesson will begin. When report time comes, the teacher types a few commands and a master progress report is generated by the computer.

But there is more potential beyond mere management. "Expert systems" can improve computer instruction. These programs maintain a profile of what the learner knows about a specific topic, and adjust the level of instruction to that needed by the student. If the student has difficulty he or she can ask the computer tutor for hints or request further explanation or a different type of explanation. Feedback can be rapid, and pacing of instruction can be adjusted appropriately.

One program can diagnosis errors and prescribe instruction in the area of substraction. This program, called "BUGGY"and written by John Seely Brown, keeps track of student errors, scrutinizes the errors a student makes in subtraction and then provides remediation for that particular error or "bug" discovered in the student's thinking. Suppose that a student makes the error of subtracting the smaller number from the larger rather than borrowing. Brown's program records the error, and checks to see if that is the real bug by giving the student similiar problems. If the hypothesis is confirmed, the program remediates the bug. Brown's program simulates some of what a teacher would do in a diagnostic/remedial model, and represents effective instruction.

However, to catch the vision is not enough. A glance at what is indeed possible is not to imply that quality programs are available commercially in many subject areas. This brings to mind several areas concerning software that need to be included in teacher training. First, educators need to be able to evaluate the software that is available. It is necessary to examine the design of software. Has it included sound educational principles regarding pacing, repetition, modeling, reinforcement, and feedback? Second, inservice and preservice training need to include provision for the review of a wide variety of software, and methods of finding a specific piece of software to meet specific needs, and then actually using software with students to see if the desired result is achieve. Third, educators need to make their standards for good software known to commercial developers. Training should be aimed at helping educators to establish those standards.

How much time should be devoted to the design and evaluation of software in our four credit hour course? Since the area is so important thirty to fourty percent would be appropriate.

There are other important areas that need to be considered in training such as problem solving environments (LOGO), learning tools such as word processors, and video disk technology. Also important are prosthetic devices such as computerized body parts, and special communication devices for various physically handicapped.

Really, I'm afraid that Rip Van Winkle would be no more safe in the special education classroom than in the department store. Computers, printers, prosthetic devices, and others defy the saying that schools never change.

THE COMPUTER: EXTENSION OF THE HUMAN MIND AND CHALLENGE TO HUMANNESS

P. Kenneth Komoski

Computer Uniqueness/Human Uniqueness

There is something quite comfortable about viewing of the computer as an extension of the human mind. It's comfortable because it allows us to view the computer as the child of our brain that can help us with the mental labors of the "information age" just as the child of our genes once helped us with the physical labors of the agricultural age.

However, this view also raises the sort of questions that tend to make people uncomfortable: An extension of the human mind for what purposes? Mental labor toward what ends? Are all human minds to have equal access to mind-extending capabilities of the computer? Should, (indeed, can) the extent of the computer's work and influence be limited? What will the relationship be between our natural mind and its artifactual extension?

Evidence that such questions make us uncomfortable can be found in our tendency not to answer them directly. What we usually do with questions like these is to deal with them indirectly by answering them through our actions, our artifacts and, especially, by how we use those artifacts in relation to ourselves, others, and our social institutions. If we are willing to recognize this tendency, it is also important to recognize that the answers we are already building into our actions <u>vis à vis</u> computers represent some of the most important, unique, and far-reaching messages we may ever send abroad into our world. I maintain that these messages are not just important, but unique, because the computer is the most general of all human artifacts. This, by definition, makes it unique.

Because of the computer's uniqueness, I somehow doubt that an analysis of our actions toward other ubiquitous but less <u>general</u> artifacts such as the automobile, the telephone, and television will provide us with much useful insight as to what purposes, ends and values we are going to build into our actions as we use and, perhaps, even fuse with the computer. The fact that the computer is effecting changes in the design, use, and future development of each of these more specialized technologies speaks to both its generalness and its uniqueness.

The degree to which the computer also promises to effect our own future development further speaks to its uniqueness. It also raises important questions about the degree to which we, as humans, will be able to sustain our own uniqueness in a generally computerized world. Thus, I am not simply engaging in word play when I speak about "using and fusing" with the computer. Nor am I merely furthering that playfulness when I say that there is a very good chance that we will not and perhaps cannot, refuse to use and fuse with the computer.

In Programs of the Brain, J.Z. Young convincingly demonstrates the extent to which we humans are organized by genetically transmitted information, developed and programmed into our DNA molecules by the workings of the evolutionary process. These ancient programs, Young tells us, are present in each of us waiting to be activated by the appropriate environmental stimuli; programs of all types that help us to have fuller lives, aesthetically, morally, spiritually, intellectually and physically. When reading Young, it's hard not to be struck by the fact that less than a decade after the development of the first modern computer, which we soon began to use to code, to organize, and to analyze the elements of our increasingly complex external environment, we also discovered that nature had long been organizing and transmitting the elements of our even more complex internal environment by means of a very similar process.

When reading Marvin Minsky's comments on artificial intelligence (especially his oft-quoted quip: "Computers may eventually keep us as pets"), one can't help feeling that the most important human task ahead of us (aside from managing to avoid destroying each ot and our life-sustaining ecosystem) will be to identify and to maintain those uniquely human qualities and characteristics that will enable us humans to differentiate ourselves by our uniqueness from our increasingly intelligent artifacts. For me, Robert Jastrow's, The Enchanted Loom: Mind In the Universe, more than any other current work, seems to confirm this.

Jastrow sees the evolution of the silicon-based computer as an inevitable and necessary extension of a human mind that is no longer able to evolve rapidly enough biologically to deal with its increasingly complex environment. He also sees -- as equally inevitable -- an evolving symbiotic relationship between the human mind and its newly-evolved silicon extension. Whether Jastrow's vision of the human mind's being able to exist after the demise of its biological host by being up-loaded into what might be described as a solar-powered space-chip is valid, remains to be seen. But Jastrow, a founder of NASA, and a scholar with impeccable scientific credentials indicates that some of us may be able to test the validity of his vision before the end of our own lives,

Jastrow views the computer not only as an extension of the human mind, but as a sort of necessary ontological intention, as well as an invention that may make it possible for us poor mortals to aspire to solar-powered, silicon immortality. All of this says to me that those important and discomforting questions raised by recognizing computers as the extension of the human mind can be reduced to: What will it mean to be "human" as we adapt to an increasingly powerful artifactual environment? Will all, or just some of us, have access to the artifactual tools required to participate in this seemingly imperative evolutionary adaptation?

It is not my intent to answer these questions in a systematic and rigorous fashion. But rather to sketch the outline of what I hope will not turn out to be a totally inaccurate map of a new and challenging territory that exists between us humans and the most human-like, the most general and unique artifact we have so far invented.

Before venturing into this uncharted territory, I want to admit publicly that I, for one, find Jastrow's, albeit reductionist-like view of what the human mind may ultimately become, intriguing. The human mind not as computer, but at one with the computer -- a silicon soul sojourning in space, free of earth-bound concerns, contemplating God, or simply eons-long inter-planetary sunsets. Will the age-old human longing for some form of existence after death, as Jastrow predicts, be translated into a side-stepping of death through an ongoing mental life? But whether or not Jastrow's vision for the hereafter is realized in the future, it seems fairly obvious that some sort of human-computer symbiosis is going on in the here-and-now. What, I'm suggesting is that we would do well to start thinking pretty hard about the implications of this evolving symbiosis. I am also urging that our thinking be informed by continuing critical observations and analysis of our actions toward our new symbiotic partner.

Symbiosis/Education/Levels of Humanness/Caring

Let me begin with an obvious observation: We have already begun to describe computers with adjectives such as "personal," "home," and "friendly." Certain software (a rather friendly term in itself) is described as being "more intimate" than other software. Is this an indication that the symbiosis is well underway or simply a way of lowering sales resistance? The answer is, "probably both." But whatever the answer, the fact is that many people are entering a new kind of reality as a result of using/fusing involvement with computer technology. To anyone who doubts this, I recommend a close reading of Pilgrim in the Micro-World by David Sudnow. This particular work deals with the all-consuming world of one video game for one person, but I suspect that it represents a genre of book that may help us explore the new territory between our own boundaries and those of the computer. An important thing to keep in mind when reading this book is that it is not describing a science-fiction fantasy, but rather providing insight into what may become for many a kind of science-friction reality that can challenge us to differentiate between our own who-ness and the that-ness of an artifact that can both absorb us and do more and more of what we once saw as unique capabilities of our autonomous human selves.

Of all human activities through which we will continuously face this challenge, education is surely one of the most important. And if our definition of education encompasses (as it must) the learning that is going to take place through our increasing use of computers not just in schools, but in our home and work environments as well, then education may well become the crucial activity through which a human-computer symbiosis will evolve. In addition, education may become the process (as C.S. Lewis, a generation ago in his The Abolition of Man, intimated it might) through which we will redefine and perhaps even define away, our humanness.

I say this not only because education is becoming more of a life-long, life-wide process, but because education has always had the paradoxical function of helping to make us different (from each other and from what we've been) through its specific branches, while it also works to make us the same (the same as each other and the same as the culture we're a part of) through its more common objectives.

The coming of the computer is subtly lifting the level of this paradoxical function of education because the computer is challenging education to take on a new set of dual objectives: (1) to make us very different from all previous human beings and (2) to make us more aware of those unique human qualities that will continue to bind us to -- and guarantee our sameness with -- all previous and future beings identifiable as human.

As these paradoxical educational goals become better understood, translated, shaped, and reshaped into educational experiences in our homes, schools, and work places, it seems inevitable that we, ourselves, are going to be reshaped in specific ways that none of us can now clearly discern. But the questions now before us, seem very clear: Are we going to make a conscious effort to choose the general parameters and direction of this reshaping, or are we going to lose a sense of human definition and direction? And, assuming we set off in the general direction of defining our humanness, will we be able to create appropriate, and specific enough educational experiences to instill and maintain a core of humanness with which to guide our lives in our artifact-dependent world? What specific learning experiences should we create? Based on what specific set of uniquely human qualities of mind and sensibility?

The answers to these questions (especially the last) are getting more and more difficult to answer as research and development on artificial intellignece (actually artifactual intelligence) continues to advance. "The elusive qualities [of humanness] most frequently proposed by the critics of computer intelligence include learning, introspection, and aesthetic feeling; all of which suggests a certain unfamiliarity with the literature. Programs already manifest these abilities to some degree." This comment by Patrick Winston was made in 1977,-- a time before much of today's more far-reaching and better supported research on artificial intelligence had been launched.

In light of this, an educational undertaking that might have once been fairly straightforward, at least in terms of its general goals and objectives, looks today, to be more like a formidable educative undertaking. By this I mean a new learning experience for those in charge of the undertaking-- an experience through which they will have to learn by discovery (will a simple process of elimination strategy do?) to identify those specific, uniquely human attributes and qualities through which we might hope to define and sustain our humanness. And having done this, to transmit those attributes and qualities by means of implicit and explicit learning opportunities in our home, school and work environments.

Part of what lies ahead for us to discover about the "elusive qualities" of humanness -- aside from the fact that such qualities really do exist -- is that these qualaities seem to exist at two distinguishable levels. At the first level I believe we will find ourselves examining qualities such as creativity, introspection, aesthetic feeling, etc. that can be, to some degree, (perhaps even to a great degree) effectively replicated and eventually initiated by the computer. However, because of our human tendency to continuously refine and further evolve these qualities, human-generated creativity, introspection, aesthetic feeling, etc. are likely to maintain an evolutionary edge on their computer-generated counterparts.

However, I think we're apt to discover something quite different when it comes to the second or deeper-level qualities. These deeper-level, radically human (i.e., root) qualities will be distinguishable from first-level, cultivated human qualities in that the computer will, at best, -- and perhaps not at all -- be able to generate only a pale semblance of such qualities. Even after generations of increasingly sophisticated computer evolution, I suspect that these radical human qualities will always remain the province and the prerogative of the human side of the evolving human-computer symbiosis. In time, we may discover that these deeper-level qualities are those which define our humanness; while the first-level are those we may use to cultivate and refine certain aspects of our humanness.

The mapping of these defining and refining human qualities, I believe, will be a challenging, chastening, rewarding and, ultimately, liberating enterprise. One that I doubt would ever be initiated by the computer, no matter how friendly it may become.

For me, the intriguingly ironic aspect of such an enterprise is that by forcing us to think about the computer, it may bring us face to face with our humanness in a way that no other enterprise could. (Do we need any more evidence of the generalness and uniqueness of this incredible artifact we have fashioned out of simple sand?)

Caring

Eventually, the computer may, indeed be used effectively to stimulate, simulate, replicate, and even initiate every first-level human quality. In time, the computer may even learn to take care of us as well as we are learning to take care of it. But I strongly suspect that the quality of computer-generated caring will never achieve the radical quality of caring, loving behavior that is genetically available to us humans as the result of our particular and unique prior evolutionary experience. As a result, whatever degree of caring may eventually be achievable by the computer, it must, by evolutionary definition, be of a radically different quality from the level of caring that is built into us humans.

An important question, then, is whether -- within an increasingly artifactual environment -- we will continue to honor and respond to this important qualitative difference. Surely there are ample indications that we humans are willingly allowing this radical attribute of our humanness to atrophy. Can human caring, which is capable of being extended beyond one's self to the love of others -- even (on the part of some of us) to strangers and enemies -- continue to be fostered and extended to its fullest in a world in which most other human capabilities are being more swiftly extended by an increasingly powerful general artifact?

There is a good deal of everyday evidence -- in addition to the systematic arguments of critics such as Ellul and Mumford -- that the quality of caring of which humans are capable is not -- and is not likely to be -- easily fostered in an increasingly technological world. The family, the school, even the work environment seem to be in such flux that minimal day-to-day human interaction and superficial communications about these interactions are all we have come to expect. Is the recent advent of "minimum competency" as a criterion for school and job performance simply formal evidence of this growing pattern? And while there are no explicitly acknowledged minimum standards for judging family life, our society's high divorce and child abuse rates amply attest to the fact that, were such standards to be made explicit, far too many "families" would fail to meet them. The fact of the matter is that when it comes to the cultivation of the human qualities of caring and loving, we seem to have become decreasingly mindful. Instead of developing and cultivating these qualities, as the core of our humanness, the evidence is that we are failing to cultivate them; assuming that these qualities will somehow flourish without appropriate stimulation. Such a view simply doesn't jibe with the evidence provided by J.Z. Young about the way the genetically encoded programs of the human mind activate our behavior and structure our response to appropriate environmental stimuli. As support, Young offers his own 30 years of research on the workings of the human brain -- as well as the growing body of similar research.

Triggering Genetic Programs.

The important point Young makes is that if these encoded programs aren't triggered by appropriate environmental stimuli, they don't run, and when this happens a person is simply rendered less than a fully-functioning human being. Even when these programs do run, it seems that the depth of their effect and the quality of their affect on human behavior depends on the appropriateness, the quantity, and the quality of the environmental stimuli available to the individual. Contrary to current developmental theory, Young maintains that these programs of the brain are capable of being triggered by appropriate stimuli at any point in our lives, rather than only during developmentally circumscribed periods.

The implications of Young's work for the issue of human caring and loving amidst an increasingly nonhuman environment are significant: the development of these unique human attributes is always possible within individual human beings. Indeed, although responses may be tougher to trigger in older i[illegible]ividuals whose early environment lacked approp[illegible]iate stimuli, th[illegible] [illegible]ntial for developing the ability t[illegible] [illegible]re a[illegible] to love [illegible] [illegible]eviously unstimulated persons seems to remain intact througho[illegible] [illegible]ife.

While those who care about the cultivation of caring and loving individuals may find this encouraging -- especially in light of our society's shift toward an increasingly older population -- the news that "you can teach an old dog new tricks", is hardly good enough to compensate for the bad news regarding early development of these human qualities implicit in current divorce and child abuse statistics.

The question of how to trigger, foster, develop, and fully cultivate these attributes so as to sustain humanness in an environment in which needed stimuli are in short supply, is an extremely important one, perhaps, the most important one of our era. How we respond to this question will very likely determine how well we will be able to maintain our human-defining attributes. It seems to me that, at present, our chances of responding to this question in ways that will sustain our own humanness and that of future generations are, at best, 50/50.

Feedback/Recursiveness/Coevolution/Humanness

In the opening [illegible]ages of his land[illegible] [illegible]ork, Cybernetics, The Human Use of Human Beings, written [illegible]hen th[illegible] [illegible]rn computer was still in its gargantuan infancy, Norbert Wiener said that he based the word "cybernetics" on the Greek word for "governor," which had suggested itself to him because of the central concept of the new science of information: the governance or control of the performance of a system by means of a continuous flow of information in the form of corrective feedback. The concept of corrective feedback as described by Wiener, was an innovation a generation ago. But because it describes such a universal truth about the workings of all systems it is difficult -- even for those of us who peopled the pre-Wiener-World -- to act as though the concept wasn't always a part of our understanding of how systems work. And although the development of the concept was most certainly closely associated with the development of computer science, the basic concept of cybernetics, feedback, is now a central concept in our understanding of all types of physical and social systems. The reason for this is clear, once we remind ourselves of the all-important dual outcome of the basic concept of corrective feedback: (1) helping the system to adapt to changes in its external environment while (2) making it possible for the system to maintain its internal integrity. Because of the science of cybernetics we know more than we once did about how all systems adapt to a changing environment, including how our own being maintains its dynamic internal equilibrium while adjusting to a changing external reality. Building on the concept of cybernetic feedback, thinkers like Gregory Bateson and others have helped to create insights into the related cybernetic concept of recursiveness and the role it plays in helping us to understand the interconnectedness of all living systems.

Recursiveness is an important extension of the basic concept of cybernetic feedback because as originally conceived, feedback was related to the effective functioning of an individual system. However, unless the concept is extended across systems it does not provide information about the inevitable inter-relatedness of living systems to each other, and to the larger ecological system to which each individual living system is related, and upon which all nonliving systems impinge. In short, by itself, cybernetic feedback does not attend to the endless series of symbiotic relationships that weave all systems into a closely knit recursive skein. It was this recursive skein that Margaret Mead referred to as the "cybernetics of cybernetics,: and what Bateson's followers have since labelled "coevolution." (Keeney, 1982)

The human mind, and therefore human beings -- all human beings -- are now of necessity coevolving with the computer. The stimulus triggering this activity is a complex environment aggravated by increased artifactual pressure on an increasingly burdened ecosystem. And this condition is further complicated by the fact that the computer has rapidly developed into a general artifact through which we are beginning to control all other artifacts -- as well as many of our own human interactions with these artifacts. This seems to be creating more and more of a human dependency on the computer rather than a controllable (coevolutionary) relationship. This has got to be a bit of a concern to us humans, particularly when the crucial function of providing corrective, adaptive, and integrity-maintaining feedback, upon which our coevolution depends, has been delegated to the artifact upon which the delegators, themselves, are becoming increasingly dependent. The chance that some important things will go unchecked, and some important imbalances may occur ought, indeed, to concern us.

Our present relationship to the invention of -- as well as our increasing dependency on -- the computer is not that unlike our invention and rapid dependency on chemical fertilizers and pesticides as a hoped-for solution to the complex problem of increasing the world's food supply to feed the world's increasing population. As food production increases, everything seems fine. But, as the recursive process of feedback works the lethal pesticides back up through the food chain to our own human system -- and the increasing demand for petrochemical fertilizers works its way back up through the world economy to our own economy -- it becomes increasingly apparent that unless we want to solve the world's population problem by reducing our own family's reproductivity through a gene pool polluted with deadly chemicals, and by means of lives lost in wars fought for control of petrochemical sources, we had better reduce our dependency on petrochemical-based agriculture.

Now, of course, our use of the computer doesn't seem to be precipitating a flow of potentially lethal information into the world's pool of intelligence which might feed its way back through a network of interconnected data banks into our families' brains. And, as silicon is plainly as plentiful as sand, we stand little chance of fighting wars over it. So, why be concerned? Shouldn't we be more concerned that there are people who feel the computer is so polluting our lives with its requirements and constraints that it may cause us to become less socially interactive and more individually isolated? But aren't these people simply holdovers from the pre-personal, pre-computer-friendly period of computer development when the threat of having everything controlled by a priest class of mainframe computer managers seemed a real danger? And isn't it our job to help these recalcitrants to see -- through the right type of computer literacy programs -- that the personal computer is here to help them do things they personally need and want to do in more effective, less time-consuming ways? Of course, all of us -- even the doubters -- need to concern ourselves with learning how to use the evolving potential of the small, personally-controlled computer to help us do everything we need and want it to do for us as effectively as possible. But we also need to be equally concerned about the possible effects on our humanness as we become more and more informed, co-formed, and reformed -- by our evolving symbiosis with the computer.

The burden of the argument being pursued here is:

(1) Because of the inescapable recursiveness of the system of which we are a part, -- and because of our increasing dependency on our artifacts (especially one as closely modeled on, and complementary to our mind as the computer) -- we are going to be shaped more readily and more quickly, by the computer than by any other of our inventions.

(2) If this is the case, we would do well to think about how much and in what direction we might want to have that shaping take place.

(3) The information through which the shaping will be effected is already entering our own external and internal environments in the form of hands-on computer use, (including computer-aided learning), mental attitudes toward computers, and many explicit and implicit messages (mostly in the form of advertising aimed at potential users of computers).

(4) It isn't a question of whether we want to do something about how computers are affecting us, our lives, and the lives of our children -- and their children, we <u>are already doing something</u>, if only by default.

A critically important thing to remember about the recursive system in which we happen to be living (and let's also remember that it's the only one available to us) is that <u>the buck doesn't stop anywhere</u>. It just keeps on turning into recursive, evolutionary <u>change</u>. The real question is: Do we care enough about the nature of this evolutionary change to make the effort to shape it?

Assuming that we do want to shape the nature of the change that is being precipitated by the computer, on what basis might we proceed? I believe that our guide for arriving at the proper basis for such an effort can be founded on the premise that the computer will eventually be capable of so much of what we once considered the inalienable province of humans, that if we, as humans value our humanness, we would be wise to identify, to proclaim, and to practice the cultivation of those qualities that we intuitively feel to be uniquely human. In short, I think our invention of the computer may well prove to be a test of our desire to sustain and cultivate humanness in the face of complex problems that might be more easily "solved" if we weren't so damned human.

I have suggested that one of the most radical, uniquely-human things about us is our having survived and evolved to our present state by having learned to care for -- and experience love toward -- our offspring, the members of our immediate family, our extended family or clan, our race, and all people, including (as in the case of such "social mutants" as Buddha, Jesus, George Fox, St. Francis, Gandhi and certain of their followers) our enemies.

I strongly suspect that the computer -- because of the promise it holds as a new and different way of reorganizing human living and learning could become a means of bringing about what John Dewey has referred to as the "change in objective arrangements and institutions" without which, he tells us, our desire for such things as the abolition of war, industrial justice, and greater equality of opportunity cannot be achieved. However, I believe also, that the computer may provide us with undeniable evidence of our society's limited understanding of what is radically unique about us as humans. By this I mean that we may end up using the computer to extend our minds so far and so frequently in directions of nonhumanness that we may ultimately render our uniquely human qualities too weak to insure the continued survival of --not only our humanness -- but ourselves.

Competitiveness/Cooperativeness/Educativeness

We could manage to do this by using the computer to further accelerate the already highly competitive patterns of behavior in our society. Indeed, we are already setting the stage for this with the basis on which many personal computers are being marketed for home, school, and business ~~use~~. Aspiring executives are being told that with a computer they'll be able to compete more effectively on the job "be the one who has the information first." Parents are being told that having a computer at home will "give your child the edge" and help to "prepare your child to compete more successfully in tomorrow's world." Such messages may very likely tell us more about what we are setting out to teach via the computer than we are likely to discover by examining the content of all the educational software currently on the market.

This is not to say that competitiveness is not a part of being human. It is. And, clearly, competition can often produce positive results not just for the individuals who win, but also for others. It does this in athletic competition by stimulating maximum performance for all members of a team, or by stimulating individuals to compete against the records of others and establish new standards for human performance.

But contrary to much popular belief, the anthropological record supports the idea that human evolution was advanced more thorugh our ancestors having learned the value of cooperative, caring behavior than through their ability to compete successfully - especially against other human beings. "Cooperation, not conflict, was evidently the selectively most valuable form of behavior for man at any stage of his evolutionary history... Without the cooperation of its members society cannot survive, and the society of man has survived because the cooperativeness of its members made survival possible" (Montagu, 1965). Recently Kenneth Boulding has pointed out that the competitive idea suggested by Darwin's metaphor "the survival of the fittest...means nothing. If we ask 'Fit for what?' the answer is 'Fit to survive'...and we knew that anyway. In reality," says Boulding, "what survives is what 'fits' into the complex structure of ecosystems. 'Survival of the fitting'," Boulding concludes, "would be a better metaphor." (Boulding, 1983)

Such insights into how we humans have survived and arrived at where we are can help us to see that, evolutionarily, we have demonstrated a good deal of talent for fitting in to our natural, artifactual, and human environments. And while doing this we have managed to keep our own humanness intact. In fact, it seems that much of that essential humanness is based on our ability to fit into our natural environment and on our ability to cooperate with and to care for each other.

Making Promises/Keeping Promises/Maintaining Balance

Surviving and evolving through cooperativeness and caring implies the making and the keeping of promises. And because of this, I submit that another uniquely human attribute is our capacity for consciously making and keeping promises (and being aware of the effect on others when we fail to do this). I have no doubt there are those who will argue that it will one day be possible to program a computer to "care" about humans, to "cooperate with" them, to make and to keep "promises," and even be "concerned" about a failure to do so. But such a program would be either a poor program or a trick. There would be no reason for a good computer program to fail to do something it had "promised" (i.e., been programmed) to do. Because of this the computer would be, logically, unable to reflect on, and to question how and why it had failed to fulfill its "promise." This specific capacity to question how and why we behave or fail to behave in certain ways is, of course, introspection. And although computers may -- we are told -- be programmed for introspection, I feel -- as stated earlier -- that human introspection may forever remain at a different qualitative level from its computerized counterpart. Thus, I suspect that were we able to master the task of teaching a computer to care, to promise, and to introspectively question and examine its performance, that performance would be of such a different quality from that of a healthy human being that the end result could never produce precisely the same quality of "humanness." Perhaps, then, we may want to consider what sort of behavior we want from the computer, and begin efforts to shape it before it gets much further along the path of shaping us.

In addition to keeping our promises, and fulfilling our promise is the issue of keeping an ever keel -- a balance between the intuitive, often ineffable sense of our humanity and the outer manifestations of our conscious purposefullness, epitomized by the computer. As Bateson puts it: "Today the purposes of consciousness are implemented by more and more effective machinery -- conscious purpose is now impowered to upset the balances of the body, of society, and of the biological world around us. A pathology -- a loss of balance is threatened" (Bateson, 1972).

Bateson also saw -- perhaps more clearly then any other modern thinker -- that when it comes to caring, loving, dependency, and other "matters of relationship" that our human capacity for "relatively unconscious" nonverbal communications is very important as a balancing agent "between self and vis-a-vis or between self and environment and that the nature of human society is such that falsifications of this discourse rapidly becomes pathogenic."

My fear is that by allowing our symbiotic discourse with the computer to be too much guided by the conscious purposiveness which through us produced it, we are stiffling the healthy functioning of the core of our humanness, leaving us unbalanced, keel-less and rutterless, here and now and ever after.

In short, I believe we would do well to examine the evolving symbiosis between ourselves and the computer, and begin to think seriously, and publicly, about whether there are things <u>we will not have the computer do for us</u>. If this seems an unreasonable or surprising proposal -- or one that may seem to fetter research and development in <u>artifactual</u> intelligence -- let me say that I offer it in the same spirit in which, in 1974, a group of leading genetic researchers proposed that a moratorium be called on research on gene transplantation until the implications could be debated and responsible public policy decided upon. (Berg, 1974)

To keep this proposal within the immediate frame of reference, I believe that rather than looking forward to a time when we might be able to program even such radical human qualities as caring, we may be better advised to decide how we can best use the computer to help keep our society's unfulfilled promises regarding such matters as equal social, educational, and economic opportunity for all its members. One way of doing this would be to institute policies and practices that would provide equity of access to computers, plus appropriate training for those families and workers who lack the financial resources to achieve ready access to the technology. To do this, we would have to develop what Dewey characterized as "a change in objective arrangements and institutions." This is never an easy undertaking. But there are a number of such changes that could be made with relative ease through new cooperative arrangements between parents, schools, computer manufacturers, and other relevant parties.

Such changes, when achieved, could help greatly in solving the equity of access problem. (Komoski, 1983) Not to do this, I believe, could be very costly. The cost might even run as high as our failing to achieve an effective society-wide symbiosis with the computer, based on humanness.

I stated at the outset that it is the tendency of us humans to implicitly demonstrate our beliefs through our actions, and the use of our artifacts, rather than by stating them explicitly. Later, I alluded to the messages we are building into our present marketing of computers and their use. I characterized these messages as being likely to foster something less than a cooperative, caring, attitude toward fellow workers, fellow learners, fellow humans. Exactly what the collective result of these messages will be is, of course, difficult to say; however, their implications seem clear enough.

We seem to be headed toward using the computer to widen the already chasm-like economic and social gap that currently exists between those with and those without in our society. I suggest that to continue to do this -- particularly in relation to education and training of the young -- is social and economic dynamite that could make the task of sustaining our humanness in an increasing artifactual, competition-filled environment just about impossible. This, of course, is a society-wide problem that goes beyond education as an institution. Yet, at present, -- in keeping with a long established response pattern -- we seem to be expecting "education" or "the school" to somehow deal with this complex social, economic, and moral problem. Such an approach to the present problem is no more likely to succeed today than it would have 100 years ago when Emile Durkheim observed: "It education is only the image and reflection of society. It imitates and reproduces the latter in abbreviated form; it does not create it. Education is healthy when people, themselves, are in a healthy state, but it becomes corrupt with them, being unable to modify itself." This observation appears -- perhaps prophetically, -- in Durkheim's classic work, Suicide. If this seems an overly depressing, overstated allusion to our society's current state of health, I suggest that it may serve as a necessary antidote to the mixture of zealotry and complacency that characterizes much of the present response to the potential impact of computers.

But whatever the reaction to Durkheim's analysis, it has an unmistakeable ring of authenticity. In addition, it contains a challenge that must be met. However, if we are to meet this challenge successfully as a society, we must prepare ourselves to make healthy responses -- generated from the core of our humanity -- to the dual task of adapting to the computer, while seeing to it that the computer adapts itself to our humanness. Unless we can bring ourselves to do this -- to make the necessary changes in our "objective arrangements and institutions" -- we will find ourselves locked into an unhealthy, unbalanced and, ultimately, dehumanizing symbiosis.

References

Bateson, Gregory. Mind and Nature. New York: Bantam Books, 1979. 253 pages.

Bateson, Gregory. Steps to an Ecology of Mind. New York: Ballantine Books, 1972. 541 pages.

Berg, P. etal. "Potential Biohazards of Recombinant DNA Molecules." Science, 185 (1974), p. 332.

Boulding, Kenneth E. "Evolution of Riches." Science Digest, 91, 3 (June 1983), pp. 30-35.

Bowen, M. Artificial Intelligence and Natural Man. New York: Basic Books, 1981. 537 pages.

Dewey, John. Human Nature and Conduct. New York: Modern Library, 1957. 302 pages.

Durkheim, Emile. Suicide. The Free Press of Glencoe, 1951. 405 pages.

Eigen, Manfred & Winkler, Ruthild. Laws of the Game. New York: Harper & Row, 1983. 337 pages.

Ellul, Jacques. The Technological Society. New York: Vintage Books. 1964. 449 pages.

Emery, F.E. Systems Thinking. New York: Penquin Books. 1978. 389 pages.

Godfrey, David & Parkhill, Douglas. Gutenberg Two. Toronto: Press Porcepic Ltd., 1982. 221 pages.

Jastrow, Robert. The Enchanted Loom: Mind in the Universe. New York: Simon and Schuster, 1981. 175 pages.

Keeney, Bradford. "What is an Epistemology of Family Therapy?" Family Process. Vol. 21 2 (June 1982) pp. 153-168.

Komoski, P.K. "What Parents Should Know obout Computers for Home and School." Network. Nat. Comm. for Citizens in Education, (March 1983), p.4.

Lewis, C.S. The Abolition of Man. New York: Collier Books, 1947. 121 pages.

Minsky, Marvin. "Steps Toward Artificial Intelligence" in Computers and Thought, Feigenbaum and Feldman (editors). New York: McGraw-Hill, 1963.

Montagu, Ashley. The Human Revolution. Cleveland: The World Publishing Co., 1965. 224 pages.

Mowshowitz, Abbe. The Conquest of Will: Information Processing in Human Affairs. Reading, Ma.: Addison-Wesley Publishing Co., 1976. 365 pages.

Sudnow, David. Pilgrim in the Micro-World. New York: Warner Communications Co., 1983. 227 pages.

Varela, J. Francisco. Principles of Biological Autonomy. New York: North Holland, 1979. 303 pages.

Wallia, C.S. Toward Century 21. New York: Basic Books, Inc., 1970. 318 pages.

Wiener, Norbert. Cybernetics. 2nd edition. Cambridge: MIT Press, 1961.

Wilford, John N. "They Dream of Racing on Sunbeams." The New York Times, Science Times, May 17, 1983, pp. C1-2.

Young, J.Z. Programs of the Brain. Fairlawn, NJ: Oxford University Press, 1978. 374 pages.

VIDEODISC/MICROCOMPUTER SYSTEMS IN INSTRUCTION

Peter Moulton

Some of the most exciting developments in the computer field are those which provide new ways for humans to interact with computers. One of the most dramatic and potentially far reaching of these developments is emerging from the combination of computers and videodisc technology. This merger promises to automate educational paradigms incorporating some of our most fundamental ways of learning -- learning through vicarious experience and learning by example. During the past 35 years television has freed much of our learning from the verbal, linear mode of the printed page and has brought about a shift in our learning/thinking processes. The microcomputer/videodisc technology promises another shift which may free the learning experience from the linear format of current audiovisual technology and provide a dynamic medium which will track and interact with the student in much the same manner as a tutor or individual teacher.

A videodisc is a device which can store and display television images and sound. In this sense it is similar in function to videotape and videocassette. The significant difference with regard to educational applications is that video material can be selected from a videodisc by positioning the read mechanism over the appropriate area of the disc rather than having to pass over some possibly lengthy section of tape. The videodisc can search out and display an individual frame or sequence of frames within a few seconds or less. In this way the videodisc provides a randomly addressable store of audiovisual information from which a person or a computer can select material relevant to a particular situation.

These features allow a student to learn by progressing through a sequence of examples, explanations, and questions selected by the computer as it assesses the student's understanding of the material. A major part of the student's learning experience can come from the presentation of relevant examples displayed on a televison screen. Learning by example -- by seeing something done or by seeing the consequences of some action -- can be an extremely effective method of learning which requires less verbal skills and vocabulary than textbooks and traditional lecturing. Although communication skills are essential, there are many areas of education in which the major effort is spent toward learning the vocabulary necessary to acquire some skill. The videodisc/microcomputer technology offers some possibilities of reducing this effort.

A videodisc/microcomputer system consists of three essential components: the hardware or the mechanical, electronic equipment, the content or the audiovisual material, and the logic or the computer program which determines how the system will interact with a student. This paper will present a brief review of the videodisc/microcomputer technology, a brief description of the productions process, and finally a discussion of the requirements for software authoring systems. (3,4)

The Hardware: Videodisc Technology

Although videodiscs have appeared on the consumer market as an alternative to videotape only in the past five years or so, disc storage for television signals dates to the earliest beginnings of television in the 1920's when the common phonograph technology was employed. The significant problem in using an audio recording technology for video signals is that the bandwidth, or the amount of information required for a video signal is aproximately 100 times greater than that needed for an audio signal. This means that even if we could get the necessary information transferred from a phonograph record rapidly enough, that we would get only one-hundredth the amount of recording time from a standard record -- perhaps only 20 or 30 seconds. In order to place enough information on a manageably sized disc to provide a reasonable length of recording time, say one-half hour, the number of tracks on the record would have to be very large and hence the size of the tracks very small, and the amount of space on a track for a single piece of information would have to be very small. This means that if a mechanical "needle" were to ride in the track detecting bumps and wiggles, that the record would be quite vulnerable to wear. Alternative technologies have been developed to cope with problems of bandwidth and wear.

There are several factors to consider in evaluating a videodisc technology for a particular application. First is the total amount of recording time on a disc. This is very important for recording feature length movies, but perhaps less so for instructional materials. Second is the ability to select a particular frame or sequence of frames on the disc and the amount of time required for random selection. This factor is of minor importance for feature length films, but is essential for interactive instructional materials. Third is the ability to display a single frame as a still image. This feature is important for instructional materials, but of minor interest for films. Fourth is the vulnerability of the disc to wear and damage from handling. A fifth factor is the ability to record as well as reproduce video signals. This is important if videodiscs are to compete with home videocassette records used to record television programs. It is of less importance to instructional applications in which precorded discs will be used in most settings.

Capacitance Discs

There are currently two technologies for videodiscs -- a capacitance technology and an optical technology. The capacitance system employs pits in a nonconducting outer layer on the videodisc which create a variance in the capacitance between the conducting stylus and the conducting middle layer of the disc. The variance in capacitance can be used to reproduce the video signal. Two companies which are using this system are RCA and JVC. The RCA system uses very fine grooves, not to carry the signal, but to guide the stylus. The JVC system does not use grooves, but uses a magnetically controlled servo mechanism to guide the stylus. For this reason it is less subject to wear. The RCA system records four video images on one track, which for a given number of tracks on a disc means that more material can be recorded than if a single frame were placed on one track. However, this also means that if the stylus were held in a fixed position, that the four frames would be repeatedly displayed in sequence and that a flickering image would result rather than a fixed image. Of course four identical frames could be placed an the one track, but this would reduce the storage capacity of the disc. The JVC system places two frames on a track.

Optical Discs

The optical technology again uses discs with pits to record the signals, but these pits are detected by a laser beam rather than a stylus. There are two dominant optical technologies. The first, the reflective discs, reflects the laser beam from a reflective coating on the pitted surface. A transparent protective coating is applied over the reflective coating. This technology wasdeveloped independently by MCA and by Philips and is currently used in a number of players such as Pioneer and Magnavox. The second, the transmissive optical disc, uses a focused laser beam which passes through the disc. The read mechanism of a optical disc never touches the disc surface which means that there is no wear. This technology was developed by Thomson-CSF. Also, the protective coating on the reflective surface makes the disc less vulnerable to dust, fingerprints, and other consequences of handling.

Optical discs employ one of two recording modes -- constant angular velocity (CAV) and constant linear velocity (CLV). CAV records one frame per track. This means that one frame is displayed for each 360 degrees rotation of the disc regardless of which track is being displayed. Since the outer tracks of the disc are longer than the inner tracks, if one frame is recorded on each track, the information on the outer tracks is recorded less densely. CLV records at a constant density, so that one frame is always recorded on a given length of track. This means that the disc must rotate more slowly when the outer tracks are being read than when the inner tracks are being read. While this allows more information to be recorded, it eliminates the possibility of displaying a still image by holding the read mechanism on a particular track.

In order to obtain standard television quality motion, 30 frames must be displayed every second. If each track holds a single frame, then the disc must be rotated 30 times per second or 1800 rpm. By moving the read head over the tracks of a CAV disc more slowly than normal, slow motion of any speed can be obtained. The current optical discs hold 54,000 tracks or 30 minutes of playing time per disc recorded in CAV mode. A CLV recording holds one hour of material. The capacitance discs and the optical discs which have been described are read only devices. The production processes for these discs involve use of a videotape recording of the original material. Once a master disc has been created, however, copies can be quickly and inexpensively made by pressing or by injection moulding. The cost of the twelve inch discs is is the neighborhood of $20.00 .

Direct read and write (DRAW) and write once and read only optical videodics systems have been developed, but to date are prohibitively expensive for most applications. Also, eraseable optical videodisc systems are being developed which make use of a plastic material in which small regions can be deformed by the heat of a laser beam. Although the magnetic recording technologies have offered a medium which can be written and erased, to date the storeage capacities of magnetic discs have been too small for all but a few applications. Recently, however, the densities of magnetic recordings have been pushed to as high as 50 million bits per square inch, and magnetic media should not be disregarded as possibilities for the future.

Videodisc Players

At the present time there are three levels of the most commonly available Videodisc player. The lowest level is primarily for playback of feature length movies in the home. These players are not capable of selecting a single track nor of displaying a still image. They cost in the range of $300 to $600. Because of their limited capabilities, they are not of great interest for educational applications. The RCA and the least expensive Pioneer players are in this category.

At the next level are players which can address and select individual frames and display still frames or slow motion. These features require CAV recording, which at present limits a disc to 30 minutes of material at normal television speed. . Such players can be useful in limited educational applications in which sections of the disk or individual images can be manually selected by entering the appropriate track number into a keyboard on the disc player or on a remote control unit. These players are not capable of accepting other information from the user and therefore are not useful for interactive applications. Such players are available in the $700 range. Pioneer is currently marketing a home player in this category.

In order to input information from the user and to make decisions regarding subsequent frames to display, some sort of computing capability is required. This capability is built into the next level of player -- the industrial quality videodisc player. These players incorporate a limited computing capability which can be programmed to input responses from the user and to use these responses to select the next frame to be displayed. They also have a limited capability to store data and to output to a printer information such as responses and scores. The programs for these built-in computers can be loaded from the videodiscs themselves. These players have been used in a number of industrial training and educational settings. They are priced in the range of $2000 to $3000 dollars. Pioneer, Sony and others are producing players of this type.

Microcomputer Interfacing

To date the computing capability built into even the most advanced commercial videodisc players is quite modest. In order to develop a sophisticated turtoring machine, much more extensive computing equipment is required. The recent advances in microcomputers have made available considerable power in small reasonably priced packages. To enable the computer to direct the actions of a videodisc player requires an interface, which is an electronic circuit designed to convert the signals output by the computer into signals that can be interpretted as commands by the videodisc player. The industrial quality players have standard computer interfaces built into them. Separate interfaces are available for players with frame selection capability.

Incorporating a microcomputer into the system provides more sophisticated computing capability, more memory to keep track of information regarding the student and the video material, and more varied input/output equipment such as printers, disk drives for the computer's storage of programs and data, and communications equipment to communicate with other computers. Several companies are now marketing complete systems consisting of microcomputers and players.(2)

There are several features of the harware of a videodisc/microcomputer system which should be considered. The computer itself will normally need to display information on a graphics screen, and having this be the same screen as the videodisc display is usually desirable. This may be a feature of the interface which is some cases may provide only a computer-operated switch which allows either the computer or the player to display on the screen, but not both simultaneously. More sophisticated interfaces will interlace the signals of the computer with those of the player allowing the computer display to overlay the videodisc image.

Other important features of the microcmputer determine how the student can input responses to the computer. All too often a student's ability to interact with the computer is hampered by a lack of typing skills. Touch panels on the video screen allow the user to manually point to features displayed on the screen and to have the location of the finger input to the computer which "knows" the location of the various features. ("knows" of course implies that someone has included this information in the computer's data or program as part of the instructional software.) A light pen serves a similar function. Speech synthesizers and voice recognition devices are now becoming available for microcomputers and promise the possibility of vocal interaction with the computer.

The Content: Producing a Videodisc

At the present time production of a videodisc entails the production of a videotape which is then mastered onto a videodisc. This process is both costly and time consuming. (7) There are several stages to this process which can roughly outlined as follows:

1. development of an overall plan for the material with an outline of specific segments
2. detailed scripting of each segment including specification of the video material and the associated audio content
3. preparation of graphics material
4. assembling actors and materials for the various segments and scenes
5. videotaping the material
6. editing the videotape as it is to be recorded on the videodisc
7. sending the videotape to a studio which produces a videodisc master and then copies the disc
8. reviewing the videodisc and obtaining frame numbers to be used by the computer program.

Several companies in this country and Japan produce the video master and the copies. The cost is approximately $3000 for a master and considerably less for each copy as the number of copies increases. The other production costs of course can vary considerably. The costs of producing a master can be expected to decrease especially as direct write video units become available. While the content of a videodisc will be determined by educators, the planning and scripting and production of the content of the disc will remain costly, time-consuming, and generally an activity for people trained in this area.

The Logic: Authoring an Interactive Program

While the hardware of the vidodisc technology is still in an early stage, it is developing rapidly, and we are certainly promised an adequate technolgy for the selection and display of audiovisual material. The methods for producing video material are well known and as long as human actors are used the methods and costs are not likely to change significantly in the near future. What does represent and will continue to represent the most serious obstacle to the effective use of this technology in the field of education is our ability to instruct computers to track students and select material with a degree of flexibility, creativity, and helpfulness approaching that of a competent tutor. (6)

We need more intelligent computer involvement in our problem solving activities. Although computers have become very powerful and influential as our intellectual slaves, to have them simply carry out our prespecified commands is no longer sufficient. Many of our problem solving activites are too complex to specify in complete detail. If we are asked to diagnose a disease and prescribe a treatment, we cannot spell out all of the possicble cases. Guidlines and heuristic principles can be descibed, however, and if a computer is to help us, it must be able to employ these guidelines to search out possible solutions. This search may involve the computer's asking us for more information and may result in the computer's discovery of possibilities which we have not considered. In this way the computer can evolve from a passive tool into an active participant in our search for solutions. A specific problem solving situation of interest to us is that of designing a sequence of instructional materials for a student. Imagine the problem solving processes of a tutor as examples, explanations, and questions are generated for a student during an hour of instruction. While a general plan for the hour may have been developed, the tutor must rely upon a repetoire of material which can be brought forth as the perceived needs of the particular student in the particular moment may demand. Only a poor tutor would rigidly adhere to a specific plan, disregarding the needs of the student as his or her grasp of the material varies. Instead, the ideal tutorial hour proceeds as a dialog between the tutor and the student.

Although computer aided instruction (CAI) has been avialable for over 20 years, all but a few experimental applications have relied upon programming methods which require that all of the possibilities for the presentation of material be explicitly specified by the developer of the material. The richness of a CAI program derives in a major part from the complexity of the possible paths through the material which of course deterine how well adapted the program will be to a particular student's needs. The network of possible paths is referred to as the <u>branching structure</u> of the program, and it is the complexity of the branching which makes the development of CAI programs so costly and which places limits upon the quality of these programs.

In order to illustrate the problems of explicitly describing the branching structure of a computer administered lesson, consider the example of teaching a student computer programming. In particular consider teaching the concept of the precedence or order of evaluation of operators within arithmetic expressions. As one small part of this example, it is important for the student to know that multiplication, indicated by the symbol '*' will be evaluated before addition. Thus, '3+5*4' evaluates to 23 rather than 32.

This material might be presented in two segments, the first teaching the order of evluation, and the second containing examples to be evaluated. We might have four frames or exercises in each segment as follows:

```
precedence segment
  P1: Which operator (+,*) will be performed first?
        Y + W * T
  P2: Which operator (+,*) will be performed first?
        Z * Y + W
  P3: Which operator will be performed first ?
        T * ( V + S )
  P4: Which operator will be performed first ?
        ( Y + T ) * Z

evaluation segment:
  E1: Given that Y=5, W=3, T=2, what is the value of
        W + T * Y
  E2: Given that X=4, Q=7, S=3, what is the value of
        X * Q + S
  E3: Given that T=2, V=5, W=6, what is the value of
        T * ( V + W )
  E4: Given that T=3, S=4, X=7, what is the value of
        ( T + S ) * X
```

The administration of each frame would involve the following sequence of steps:

1) display the frame

2) input the response
3) if response is correct, take correct action
4) if response is incorrect, take remedial action

In the simplest case the "correct action" would be to proceed to the next frame, and the "remedial action" would be to display the correct answer. More sophisticated procedures might request that the student retry for an incorrect responsse, and might recorde history of responses in order to jump ahead when a student obviously knows the material. This pattern of administration leads to the following control stucture:

```
     P1
    /   \
false    true
          :
          P2
         /  \
    false    true
              :
              P3
             /  \
        false    true
                  :
                  P4
                 /  \
            false    true
                      :
                      E1
                     /  \
                false    true
                          :
                          E2
                         /  \
                    false    true
                              :
                              E3
                             /  \
                        false    true
                                  :
                                  E4
                                 /  \
                            false    true
```

Notice how cumbersome the branching becomes even with a trival example. This amount of programming not only becomes costly, but also introduces many possibilities for error.

During the past seven or eight years developments in the field of artificial intelligence have produced a number of expert systems or knowledge based systems. (5) They are called expert systems because they perform problem solving activities in difficult areas using methods derived from expert behavior in these fields. They are called knowledge based because they incorporate large bodies of information. In general these systems are characterized by the need for large numbers of complexly organized possibilities, the need to formulate and test hypotheses and partial solutions, and the need to work with situations which change over time. One of the earliest and most widely publicized expert systems is the MYCIN system developed at Stanford in the late 1970's.(1) This system operates as a consultant in the diagnosis and treatment of diseases of the blood. It is reported to function as well as an expert in the field and better than a physician not working in this area. Features of the system are the formulation of hypotheses, the ability to request information from the physician, and the ability to describe its processes in reaching certain conclusions or in rejecting certain possibilities.

The problem solving tasks in an instructional setting focus on the selection of relevant material to present to the student in light of the student's changing knowledge, understanding, and performance and in light of all of the possible examples, explanations, and questions which might be presented to the student.

Many of the expert systems make use of rule based or production based languages. These languages allow the programmer to specifiy conditions which when met will invoke the execution of some action. In the case of CAI applications, the conditions conditions specify attributes of the student to be matched with attributes describing frames of material to be presented. Student attributes could include information regarding the level of the student and the student's progress in the current lesson. The attributes describing frames of material could specify the level of the material, subject areas, and prerequisites for the presentation of a particular frame. The action to be performed for a given match of student and frame could include the presentation of material, the input of a response from the student, and the modification of attributes describing the student.

This kind of organization of a CAI system could be conceptualized as a pool of frames of material, a set of attributes describing a student and his or her performance, and a set of conditions/actions which match the current state of the student with one or more frames. Each match results in a candidate for presentation to the student. When more than one such candidate frame is available, a selection can be made either randomly or according to some prespecified criteria. The attributes of the student are modified by the actions associated with the conditions, thereby keeping track of the progress of the student. Graphically such a system can be pictured as follows:

```
 ------------              ------------            ------------
| pool of     |           | conditions |          | student     |
| frames      |---->      |   with     | <----    | attributes  |
| with        |           | actions    |          |             |
| attributes  |           |            |          |             |
 ------------              ------------            ------------
                               ||
                               ||
                               \/
                           -------------
                          | candidate     |
                          | frames for    |
                          | presentation  |
                           -------------
```

In order to compare this method with that requiring specification of all of the branching, consider the following example which illustrates the way in which the conditions and actions might be written in a rule based language designed for this purpose. Here the portion of the statement following the word IF is the condition and the portion following the arrow is the action. In this simple example, the attributes of the student include topic currently being worked on and the number of correct responses in this topic. The attributes of the frames include subject areas.

```
IF [ student working on arithmetic precedence
     AND student answered less than 3 correct
     AND frame major topic is arithmetic expressions
     AND frame subtopic is precedence ]
->   present frame, input response, update student
     attributes.

IF [ student working on arithmetic precedence
     AND student answered 3 precedence questions
         correctly
     AND student answered less than three precedence
         calculation
     AND frame major topic is arithmetic expressions
     AND frame sub topic is precedence calculation ]
->   present frame, input response, update student
     attributes.
```

In this approach, the number and complexity of the condition statements is not directly related to the amount of material or the number of frames within the system. In other words, we might double the number of frames without doubling the number of branching statements

A number of languages have been developed for rule based systems. The system developed for MYCIN has been employed as a general consulting system package. To the author's knowledge no language has yet been developed for CAI applications.

While an expert system approach to CAI using videodisc/microcomputer systems has promise for offering very powerful instructional tools, the cost would not be cheap. Such a system is something which only a computer expert working with expert teachers could produce. In order to enable the computer to perform as an expert, much of the knowledge of the expert teacher must be put into the computer.

In addition to computer systems facilitating the development of the logic of the system, other computer tools are needed. In particular support is needed to develop computer generated graphic displays and to handle a variety of modes of student input. Also, computer editing systems are needed to aid the developers in the review of the videodisc and the determination of frame numbers to be associated with the logic of the system.

Conclusions

The videodisc/microcomputer technology promises exciting and powerful educational systems which will interact with students with the expertise of an individual tutor and with all of the interest and attraction of television and video games. These systems will be able to track a student's performance and use this information together with the student's history to select relevant audiovisual examples, explanations, and questions to present to the student. Although these systems are just now emerging, they are developing rapidly. While the cost of the hardware is already low and will become even more reasonable, the cost and time required for the production of the video material and the logic of the computer programs will be high. The major obstacle to realizing the full potential of these systems lies in the development of the logic of the computer software. The recent developments in expert systems and knowledge based systems offer some direction in overcoming this obstacle.

References

1. Davis, R., Buchanan, B., and Shortliffe, E. "Production Rules as a Representation for a Knowledge-Based Consultation Program." Artificial Intelligence, 8(1977), pp.15-45
2. Paris, J. "The Interfaces" Popular Computing 2,6 (April 1983) pp.83-84
3. Schneider, E. and Bennion J. Videodiscs Educational Technology Publications, Englewood Cliffs, New Jersey 1981

4. Sigel,E.,Schubin,M.,Merrill,P. Video Discs
Van Nostrand Reinhold, New York 1980
5. Stefik,M.,Aikins,J.,Balzer,R.,et.al. "The Organization of Expert Systems, A Tutorial." Artificial Intelligence,18(1982), pp.135-173
6. Thorkildsen,R. "Microcomputer/videodisc Authoring System for Instructional Programming." Annual Meeting of the American Educational Research Association, March 1982
7. van Melle, Wm., "A Domain-Independent Production-Rule System for Consultation Programs." Proc.Sixth International Joint Conference on Artificial Intelligence, 1979
8. Wicat Systems, "Interactive Videodisc Design and Production."BYTE 7,6 (June 1982) pp.56-57

MICROCOMPUTER APPLICATIONS IN EDUCATIONAL MANAGEMENT:

A REVIEW OF INTEGRATED MANAGEMENT SOFTWARE

Philip K. Piele
Professor and Director
ERIC Clearinghouse on Educational Management
College of Education
University of Oregon

Outline of Presentation

A. Introduction

B. First Generation Management Applications: Single-Task Software

1. Primary Applications

   a. Word Processing
   b. Spreadsheet
   c. File Management
   d. Business Graphics

2. Secondary Applications

   a. Telecommunications
   b. Local Area Network (LAN) Support
   c. Project Management

C. Second Generation Management Applications: Multiple-Task (Integrated) Software

1. Integrating Programs

   Memory/Shift

2. Keyboard-Oriented Integrated Programs

   a. File-Based Packages

      DBM II

   b. Spreadsheet-Based Packages

      MBA
      1, 2, 3

3. Screen-Oriented Integrated Programs

a. Combined Software-Hardware Packages

Lisa (Local Integrated Software Architecture)

b. Software Only Packages

VisiOn

D. SUMMARY, RECOMMENDATIONS, AND FUTURE DEVELOPMENTS

Selected References

Mello, Jr., J. "Lisa: Apple Introduces State of the Art in User Friendliness," Apple Insider (April 1983), pp. 19-20.

Miller, H. "The Spreadbase Race," PC World, 1, 1 (February 1983), pp. 102-117.

Needle, D. "Integrate DBM II with Other Software," InfoWorld, 5, 18 (May 1983).

________. "New Product Lets You Integrate All Your Applications Software," InfoWorld, 5, 18 (May 1983).

"New Software Take the Drudgery Out," Business Week (December 13, 1982), pp. 73-74.

Williams, G. "Lotus Development Corporation's 1-2-3," BYTE (December 1982), pp. 182-197.

Williams, A., "The Graphsheet Contenders," PC World, 1, 1 (February 1983), pp. 124-133.